Tianbao Yang

Compositional Optimization for Advanced Machine Learning

To my parents, the foundation of my values; to my wife, my steadfast companion; and to our children Evan and Eileen, the joy of our lives.

Preface

知者行之始，行者知之成
—王陽明
"The best theory is inspired by practice.
The best practice is inspired by theory."
— Donald Knuth

Optimization is central to machine learning (ML), which in turn forms the foundation of artificial intelligence (AI). From training deep neural networks to fine-tuning Large Language Models, almost every advancement in AI relies on solving some form of optimization problem. While classical methods based on empirical risk minimization (ERM) have powered much of early progress in ML, they are no longer sufficient to address the growing complexity of today's AI challenges. This book aims to bridge that gap by offering a systematic treatment of the emerging optimization paradigm known as **compositional optimization** and its applications in modern AI. Many critical optimization problems in ML now exhibit intricate compositional structures as $f(g)$ or $\sum_{i=1}^{n} f_i(g_i)$ that go beyond traditional frameworks, where both f and g are non-linear functions and potentially non-convex, extending beyond the scope of traditional optimization paradigms. However, most existing texts remain focused on classical stochastic optimization and ERM, overlooking the depth and diversity of these newer challenges.

Motivation of writing the book

Optimization once held a central spotlight at leading ML venues such as NeurIPS and ICML. In recent years, however, the field has seen an influx of new topics in AI, capturing the interest of students and early-career researchers. While attention has increasingly shifted toward foundation models and AGI, the importance and impact of optimization remain as vital as ever.

As someone working at the intersection of optimization and machine learning, I feel a dual responsibility. **First,** to bring cutting-edge optimization techniques to the

broader ML/AI community. When I speak with researchers in ML/AI and mention my focus on optimization for machine learning, I am often met with questions like, "*What problems are you working on?*" or "*Are these theories truly useful, given that they rely on assumptions that may not be easily verified in practice?*" Some even remarked that optimization's only practical contribution to AI is the Adam algorithm. This reflects a common misconception that optimization in ML is limited to training algorithms like SGD or Adam, which is far from the truth. **Second,** I feel a responsibility to encourage researchers in mathematical optimization to engage more deeply with the challenges of modern AI. Many researchers in traditional optimization are eager to contribute, but the rapid pace of AI along with the constant influx of new models and terminology can make it difficult to identify core problems where optimization insights are most needed. Working at this intersection gives me a unique perspective: recognizing fundamental challenges in modern AI, such as the training of large foundation models, and abstracting them into rigorous mathematical frameworks where optimization methods can offer meaningful solutions. I hope this book contributes to bridging the gap between the AI and optimization communities and inspires new collaborations across these fields.

At first glance, the focus on compositional optimization in this book may seem narrow, but it is deeply connected to fundamental learning and optimization principles including discriminative learning and robust optimization, and has broad applicability across ML and modern AI, which will be shown in this book. In particular, this book introduces a new family of risk functions termed X-risks, in which the loss function of each data involves comparison with many others. We formulate empirical X-risk minimization as finite-sum coupled compositional optimization (FCCO) - a new family of compositional optimization. After five years of intensive research on this subject, we have explored different aspects of FCCO, from upper bounds to lower bounds, from smooth objectives to non-smooth objectives, from convex problems to non-convex problems, and from theoretical complexity analysis to applications in training large foundation models. While significant progress has been made, many open questions remain. Nevertheless, we believe it is time to share this advanced body of knowledge with the broader community in the form of a comprehensive book.

Structure of the book

This book is crafted to engage both theory-oriented and practice-driven audiences. It presents rigorous theoretical analysis with deep insights, complemented by practical implementation tips, Github code repositories, and empirical evidence—effectively bridging the gap between theory and application. It is intended for graduate students, applied researchers, and anyone interested in the intersection of optimization and machine learning. The readers are assumed to have some basic knowledge in ML. The materials in this book have been used in my graduate-level course on stochastic optimization for ML.

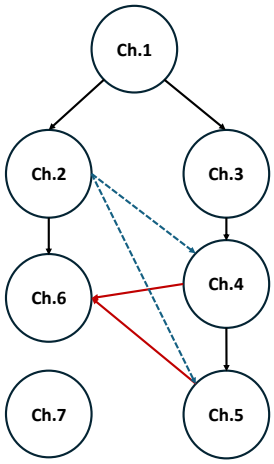


Fig. 0.1: Structure of the Book Chapters. Dashed lines indicate motivation. The red solid lines indicate application. Other solid lines indicate dependency.

The book is organized as follows. Chapter 1 reviews the fundamentals of convex optimization essential for the material presented in this book. Chapter 2 introduces advanced learning methods that go beyond traditional ERM framework so as to motivate compositional optimization. Chapter 3 presents classical stochastic optimization algorithms and their complexity analysis in both convex and non-convex settings. Chapter 4 delves into stochastic compositional optimization (SCO) problems with algorithms and complexity analysis. Chapter 5 explores algorithms and analysis for solving FCCO problems. Chapter 6 presents applications of SCO and FCCO in supervised and self-supervised learning for training predictive models, generative models, and representation models. Chapter 5 and 6 are largely devoted to the original research conducted by the author and his team. The dependencies and flow among the chapters are illustrated in Figure 0.1. Practitioners may focus on Chapter 2 and Chapter 6. For theory-oriented audiences who are interested in ML applications, I strongly recommend reading Chapter 2 and Chapter 6 as well.

Acknowledgments

This book would not have been possible without the dedication and contributions of my students. I would like to thank my former and current Ph.D. students, visiting students and postdoc who contributed to both the theoretical and empirical results presented in the book. In particular, I acknowledge significant theoretical contributions from Bokun Wang, Quanqi Hu, Zhishuai Guo, Wei Jiang, Yan Yan, Qi Qi, Ming Yang, Xingyu Chen, Yao Yao, Yi Xu, Mingrui Liu and Linli Zhou, and significant empirical contributions from Zhuoning Yuan, Gang Li, Xiyuan Wei, Dixian Zhu, Siqi Guo, Zihao Qiu, Vicente Balmaseda and Anant Mehta. Special thanks go to

Bokun Wang for his help on preparation of the lecture notes for my course with the initial version of proofs of many methods covered in this book. I thank Ning Ning for proofreading some chapters.

I am grateful to my academic collaborators Qihang Lin, Yiming Ying, Lijun Zhang, Tuo Zhao, Yunwen Lei, Shuiwang Ji, Nitesh Chawla, Zhaosong Lu, Jiebo Luo, Xiaodong Wu, My T. Thai, Milan Sonka, Zhengzhong Tu, Tomer Galanti, Yinbing Liang, Hongchang Gao, Bang An, Ilgee Hong, Guanghui Wang, Limei Wang, Youzhi Luo, Haiyang Yu, and Zhao Xu, and industrial collaborators Rong Jin, Wotao Yin, Denny Zhou, Wei Liu, Xiaoyu Wang, Ming Lin, Liangliang Cao, Xuanhui Wang, Yuexin Wu, and Xianzhi Du. I would like to thank my long-term collaborator Qihang Lin. We have worked together on the application of FCCO to constrained optimization featured in the book. Special thanks to Guanghui Lan, Chih-Jen Lin and Stephen Wright for their insightful discussions on subjects covered in this work. They have inspired me to solve some of the fundamental optimization problems covered in this book. I am especially thankful to My T. Thai for encouraging me to publish this book.

I owe a great debt of gratitude to my PhD advisor, Dr. Rong Jin, who introduced me to the world of optimization and taught me the value of focus.

I am deeply grateful to my department head, Scott Schaefer, as well as to all my colleagues in the Department of Computer Science and Engineering, for fostering such a positive and collaborative atmosphere. I also like to thank my former colleagues at the University of Iowa.

Finally, I am grateful for support from the National Science Foundation for my research under my career award #1844403, the RI core grant #2246756, and the FAI grant #2246757.

College Station, TX, USA, Tianbao Yang
January, 2026

Contents

Chapter 1
Basics: Convex Optimization

Abstract This chapter provides a concise introduction to foundational concepts in convex optimization, including convex sets and functions, Fenchel conjugates, Lagrangian duality, and the Karush-Kuhn-Tucker (KKT) conditions. Definitions are accompanied by illustrative examples to build intuition and support practical understanding. While convex optimization is a rich and expansive subject that merits its own dedicated volume, our focus is intentionally selective. We present only the essential tools and results that the author considers most relevant for understanding and analyzing optimization problems encountered in later chapters. The goal is to equip readers with a practical yet rigorous foundation, enabling them to appreciate the theoretical underpinnings of algorithm design and analysis in subsequent chapters.

Convex Optimization is the foundation of foundations!

Contents

1.1 Notations and Definitions

This book uses the following notations.

- Let us denote by $\|\cdot\|_2$ the Euclidean norm, and by $\|\cdot\|$ a general norm.
- For a differentiable function f, let $\nabla f(\mathbf{x})$ denote its gradient at $\mathbf{x}$, and $\partial f(\mathbf{x})$ denote its subdifferential set at $\mathbf{x}$.
- Let $\partial_1 f(\mathbf{w}, \mathbf{u})$ and $\partial_2 f(\mathbf{w}, \mathbf{u})$ denote the partial subgradients of f with respect to the first variable $\mathbf{w}$ and the second variable $\mathbf{u}$, respectively.
- Define the d-dimensional probability simplex as

$$\Delta_d = \left\{ \mathbf{x} \in \mathbb{R}^d : x_i \geq 0 \ \forall i, \ \sum_{i=1}^{d} x_i = 1 \right\}.$$

- Let $\mathbb{I}(\cdot)$ denote the standard indicator function, which returns 1 if the input condition is true and 0 otherwise. Let $\mathbb{I}_{0-\infty}(\cdot)$ denote the zero-infinity indicator function, which returns 0 if the input condition is true and ∞ otherwise.
- Denote by $\mathbf{1}$ a vector of all ones. Let $\mathbf{e}_i$ denote the standard basis vector with a 1 in the i-th coordinate and 0 in all other entries.
- Let $\mathbf{x} \sim \mathbb{P}$ denote a random variable that follows a distribution $\mathbb{P}$.
- $[n]$ denotes the set of all integers from 1 to n, i.e., $[n] = \{1, \ldots, n\}$.
- We use $\langle \mathbf{x}, \mathbf{y} \rangle$ interchangeable with $\mathbf{x}^\top \mathbf{y}$ to denote the inner product of two vectors.
- $\log(x)$ is in the base of natural constant e.
- w.r.t is short for with respect to.
- s.t. is short for subject to.

Definition 1.1 (Dual Norm) Let $\|\cdot\|$ be a norm on $\mathbb{R}^d$, then its dual norm $\|\cdot\|_* : \mathbb{R}^d \to \mathbb{R}$ is defined as

$$\|\mathbf{y}\|_* := \sup\{\mathbf{x}^\top \mathbf{y} : \|\mathbf{x}\| \leq 1\}.$$

Examples

Example 1.1. *$\|\cdot\|_2$ is the dual norm of itself as $\mathbf{x}^\top \mathbf{y} \leq \|\mathbf{x}\|_2 \|\mathbf{y}\|_2$.*

Example 1.2. *$\|\cdot\|_\infty$ and $\|\cdot\|_1$ are dual norms of each other as $\mathbf{x}^\top \mathbf{y} \leq \|\mathbf{x}\|_1 \|\mathbf{y}\|_\infty$.*

Example 1.3. *Let $\|\mathbf{x}\|_A = \sqrt{\mathbf{x}^\top A \mathbf{x}}$, where $A \succ 0$ is a positive definite matrix. Then $\|\mathbf{y}\|_* = \sqrt{\mathbf{y}^\top A^{-1} \mathbf{y}}$. This is because that $\mathbf{x}^\top \mathbf{y} = \mathbf{x}^\top A^{1/2} A^{-1/2} \mathbf{y} \leq \|A^{1/2}\mathbf{x}\|_2 \|A^{-1/2}\mathbf{y}\|_2 \leq \|A^{-1/2}\mathbf{y}\|_2$.*

Definition 1.2 (Convex set) A set C is convex if the line segment between any two points in C lies in C, i.e. $\forall \mathbf{x}_1, \mathbf{x}_2 \in C, \forall \theta \in [0, 1]$,

$$\theta \mathbf{x}_1 + (1 - \theta)\mathbf{x}_2 \in C.$$

Definition 1.3 (Convex function) A function $f(\cdot) : \mathbb{R}^d \mapsto \mathbb{R}$ is convex if its domain $\mathrm{dom}(f)$ is convex and

$$f(\theta \mathbf{x} + (1-\theta)\mathbf{y}) \leq \theta f(\mathbf{x}) + (1-\theta) f(\mathbf{y}), \forall \mathbf{x}, \mathbf{y} \in \mathrm{dom}(f), \theta \in [0,1].$$

It is strictly convex if strict inequality holds whenever $\mathbf{x} \neq \mathbf{y}$ and $\theta \in (0,1)$.

This inequality implies that the graph of a convex function lies below the straight line connecting any two points on the graph—like a bowl: if you place a chopstick across its edges, it will stay above the surface of the bowl.

Lemma 1.1 (First-order condition) *Suppose f is differentiable (i.e., its gradient ∇f exists at each point in* dom f*). Then f is convex if and only if* dom f *is convex and*

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) \tag{1.1}$$

holds for all $\mathbf{x}, \mathbf{y} \in$ dom f.

Proof. We first prove for one-dimensional convex function $\phi(\cdot) : \mathbb{R} \to \mathbb{R}$, we have

$$\phi(t) \geq \phi(s) + \phi'(s)(t-s). \tag{1.2}$$

According to the definition of convexity, we have

$$\phi(t) \geq \phi(s) + \frac{\phi(s + \alpha(t-s)) - \phi(s)}{\alpha}.$$

Taking the limit $\alpha \to 0$ yields (1.2).

($\Rightarrow$) Assume f is convex and differentiable on the open convex set dom f. Fix $\mathbf{x} \in$ dom f and any $\mathbf{y} \in$ dom f. Define $\phi : [0,1] \to \mathbb{R}$ by

$$\phi(t) = f\big(\mathbf{x} + t(\mathbf{y} - \mathbf{x})\big).$$

Since f is convex and the map $t \mapsto \mathbf{x} + t(\mathbf{y} - \mathbf{x})$ is affine, ϕ is a convex function on $[0,1]$. For a convex (one-dimensional) differentiable function, we have proved that

$$\phi(1) \geq \phi(0) + \phi'(0)(1-0).$$

By the chain rule,

$$\phi'(0) = \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}).$$

Thus

$$f(\mathbf{y}) = \phi(1) \geq \phi(0) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) = f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}).$$

($\Leftarrow$) Assume dom f is convex and for all $\mathbf{x}, \mathbf{y} \in$ dom f,

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}).$$

Take any $\mathbf{x}, \mathbf{y} \in$ dom f and $\theta \in [0,1]$, and set $\mathbf{z} = \theta \mathbf{x} + (1-\theta)\mathbf{y} \in$ dom f. Apply the assumption with $(\mathbf{x}, \mathbf{z})$ and $(\mathbf{y}, \mathbf{z})$:

$$f(\mathbf{x}) \geq f(\mathbf{z}) + \nabla f(\mathbf{z})^\top (\mathbf{x} - \mathbf{z}), \qquad f(\mathbf{y}) \geq f(\mathbf{z}) + \nabla f(\mathbf{z})^\top (\mathbf{y} - \mathbf{z}).$$

Multiply the first by θ and the second by $(1-\theta)$ and add:

$$\theta f(\mathbf{x}) + (1-\theta) f(\mathbf{y}) \geq f(\mathbf{z}) + \nabla f(\mathbf{z})^\top \big(\theta(\mathbf{x} - \mathbf{z}) + (1-\theta)(\mathbf{y} - \mathbf{z})\big).$$

Since $\theta(\mathbf{x} - \mathbf{z}) + (1-\theta)(\mathbf{y} - \mathbf{z}) = 0$, we get

$$f(\mathbf{z}) \leq \theta f(\mathbf{x}) + (1-\theta) f(\mathbf{y}),$$

i.e., $f(\theta\mathbf{x} + (1-\theta)\mathbf{y}) \leq \theta f(\mathbf{x}) + (1-\theta) f(\mathbf{y})$. Hence f is convex. □

Definition 1.4 (Subgradient) For a non-differentiable convex function f, let the subgradient of f at $\mathbf{x}$ be denoted by $\partial f(\mathbf{x})$, which consists of all vectors $\mathbf{v}$ satisfying:

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \mathbf{v}^\top (\mathbf{y} - \mathbf{x}), \forall \mathbf{x}, \mathbf{y} \in \mathrm{dom}(f).$$

Without causing any confusion, we often write

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \partial f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}), \forall \mathbf{x}, \mathbf{y} \in \mathrm{dom}(f),$$

where $\partial f(\mathbf{x})$ refers to some specific element of the subgradient set.

Examples

Example 1.4. $f(x) = [x]_+ = \max(0, x)$. *At* $x = 0$ *it has a subgradient* $\partial f(0) = \{\xi \in [0,1]\}$, $\partial f(x) = 1, \forall x > 0$, *and* $\partial f(x) = 0, \forall x < 0$.

Definition 1.5 (Strongly Convex Function) A function $f(\cdot) : \mathbb{R}^d \mapsto \mathbb{R}$ is called μ-strongly convex with respect to a norm $\|\cdot\|$ if there exists a constant $\mu > 0$ such that for any $\mathbf{x}, \mathbf{y}$ and $\mathbf{v} \in \partial f(\mathbf{x})$ we have

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \mathbf{v}^\top (\mathbf{y} - \mathbf{x}) + \frac{\mu}{2}\|\mathbf{x} - \mathbf{y}\|^2.$$

Examples

Example 1.5. *The function* $f(\mathbf{x}) = \frac{1}{2}\|\mathbf{x}\|_2^2$ *is 1-strongly convex with respect to the Euclidean norm* $\|\cdot\|_2$. *This follows directly from the identity:*

$$\frac{1}{2}\|\mathbf{y}\|_2^2 = \frac{1}{2}\|\mathbf{x}\|_2^2 + \mathbf{x}^\top (\mathbf{y} - \mathbf{x}) + \frac{1}{2}\|\mathbf{x} - \mathbf{y}\|_2^2,$$

which satisfies the definition of strong convexity with parameter 1.

Definition 1.6 (Smooth function) A function $f : \mathbb{R}^d \mapsto \mathbb{R}$ is called L-smooth with respect to a norm $\|\cdot\|$ if it is differentiable and its gradient is L-Lipchitz continuous, i.e., there exists a positive real constant L such that, for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$, we have $\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|_* \leq L\|\mathbf{x} - \mathbf{y}\|$, or equivalently,

$$|f(\mathbf{x}) - f(\mathbf{y}) - \nabla f(\mathbf{y})^\top (\mathbf{x} - \mathbf{y})| \leq \frac{L}{2}\|\mathbf{x} - \mathbf{y}\|^2. \tag{1.3}$$

Definition 1.7 (Bregman Divergence) Let $\varphi : \Omega \to \mathbb{R}$ be be a continuously-differentiable, strictly convex function defined on a convex set Ω, the Bregman divergence induced by $\varphi(\cdot)$ is defined as

$$D_\varphi(\mathbf{x}, \mathbf{y}) := \varphi(\mathbf{x}) - \varphi(\mathbf{y}) - \nabla\varphi(\mathbf{y})^\top (\mathbf{x} - \mathbf{y}).$$

Examples:

Example 1.6 (Euclidean distance). $\varphi(\mathbf{x}) = \frac{1}{2}\|\mathbf{x}\|^2$ *induces the Euclidean distance:*

$$D_\varphi(\mathbf{x}, \mathbf{y}) = \frac{1}{2}\|\mathbf{x}\|_2^2 - \frac{1}{2}\|\mathbf{y}\|_2^2 - \mathbf{y}^\top(\mathbf{x} - \mathbf{y}) = \frac{1}{2}\|\mathbf{x} - \mathbf{y}\|_2^2. \tag{1.4}$$

Example 1.7 (Kullback–Leibler (KL) divergence). $\varphi(\mathbf{x}) = \sum_{i=1}^d x_i \log x_i$ *for* $\mathbf{x} \in \Delta_d$ *induces the Kullback–Leibler (KL) divergence:*

$$D_\varphi(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^d x_i \log x_i - \sum_{i=1}^d y_i \log y_i - \sum_{i=1}^d (\log y_i + 1)(x_i - y_i) = \sum_{i=1}^d x_i \log \frac{x_i}{y_i}. \tag{1.5}$$

Example 1.8 (Itakura–Saito distance). $\varphi(\mathbf{x}) = -\sum_{i=1}^d \log x_i$ *for* $\mathbf{x} > 0$ *induces the Itakura–Saito distance:*

$$D_\varphi(\mathbf{x}, \mathbf{y}) = -\sum_{i=1}^d \log x_i + \sum_{i=1}^d \log y_i + \sum_{i=1}^d \frac{1}{y_i}(x_i - y_i) = \sum_{i=1}^d \frac{x_i}{y_i} - \sum_{i=1}^d \log \frac{x_i}{y_i} - 1. \tag{1.6}$$

1.2 Verification of Convexity

In practice, directly applying the definition of convexity or verifying the first-order condition of convexity can be challenging when proving that a function is convex. The following rules offer practical tools to simplify the verification process.

Second-order Condition for Twice Differentiable Functions

If a function $f(\mathbf{x})$ is twice differentiable, then it is convex if and only if

$$\nabla^2 f(\mathbf{x}) \succeq 0, \quad \forall \mathbf{x},$$

i.e., its Hessian is positive semidefinite everywhere.

Examples

We can use the above rule to verify the convexity of the following functions.

Example 1.9 (Log-Sum-Exp Function).

$$\ell(\mathbf{y}) = \log\left(\sum_{i=1}^{K} \exp(y_i)\right), \quad \mathbf{y} \in \mathbb{R}^K.$$

Its Hessian matrix is given by

$$H = diag(\mathbf{p}) - \mathbf{p}\mathbf{p}^T,$$

where $\mathbf{p}$ *is the vector of softmax probabilities with components* $p_i = \frac{\exp(y_i)}{\sum_{k=1}^{K} \exp(y_k)}$. *It is positive semidefinite as* $\mathbf{v}^\top H\mathbf{v} = \sum_{i=1}^{K} p_i v_i^2 - (\sum_{i=1}^{K} p_i v_i)^2 \geq 0$ *due to Cauchy-Schwarz inequality.*

Example 1.10 (Negative entropy).

$$\varphi(\mathbf{p}) = \sum_{i=1}^{n} p_i \log p_i$$

where $\mathbf{p} \in \Delta_n = \{\mathbf{q} : \sum_{i=1}^{n} q_i = 1, q_i \geq 0, \forall i\}$ *is a probability vector. Its Hessian matrix is*

$$H = diag(1/\mathbf{p})$$

is positive definite.

Operations that Preserve Convexity

The following operations preserve convexity:

- **Affine Composition:** If f is convex, then $f(A\mathbf{x} + \mathbf{b})$ is convex for any matrix A and vector $\mathbf{b}$.
- **Non-Negative Weighted Sums:** If f_i is convex for all i, and $\alpha_i \geq 0$, then

  $$f(\mathbf{x}) = \sum_i \alpha_i f_i(\mathbf{x})$$

  is convex.
- **Pointwise Maximum:** If $g(\mathbf{x}, \mathbf{y})$ is convex in $\mathbf{x}$ for all $\mathbf{y}$, then

  $$f(\mathbf{x}) = \max_{\mathbf{y}} g(\mathbf{x}, \mathbf{y})$$

is convex.

- **Function Composition:** The composition $h(\mathbf{x}) = f(g(\mathbf{x}))$ is convex if one of the following holds:
  - f is convex and non-decreasing, and $g(\mathbf{x})$ is convex.
  - f is convex and non-increasing, and $g(\mathbf{x})$ is concave.

  To quickly verify this, we compute the Hessian matrix assuming that both f and g are twice-differentiable:

$$\nabla^2 h(\mathbf{x}) = f'(g(\mathbf{x}))\nabla^2 g(\mathbf{x}) + f''(g(\mathbf{x}))\nabla g(\mathbf{x})\nabla g(\mathbf{x})^\top,$$

  which is positive semi-definite under either of the above two conditions.

1.3 Fenchel Conjugate

Let $f : \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$ be a proper convex function. Its **Fenchel conjugate** (also called the convex conjugate) is defined as:

$$f^*(\mathbf{y}) = \sup_{\mathbf{x} \in \mathrm{dom}(f)} \left\{ \mathbf{x}^\top \mathbf{y} - f(\mathbf{x}) \right\},$$

where the domain of the conjugate function consists of $\mathbf{y} \in \mathbb{R}^d$ for which the supremum is finite. From the definition of conjugate function, we immediately obtain the inequality

$$f(\mathbf{x}) + f^*(\mathbf{y}) \geq \mathbf{x}^\top \mathbf{y}, \forall \mathbf{x}, \mathbf{y}.$$

This is called Fenchel's inequality. If f is proper, convex, and lower semicontinuous, then the conjugate of the conjugate of a convex function is the original function, i.e., $(f^*)^* = f$.

Definition 1.8 (Legendre function) Let $f : \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$ be a proper, lower semicontinuous, convex function with $\mathrm{int}(\mathrm{dom}\, f) \neq \emptyset$. The function f is called a *Legendre function* if it satisfies:

(i) f is differentiable on $\mathrm{int}(\mathrm{dom}\, f)$, and for any sequence $\{\mathbf{x}_k\} \subset \mathrm{int}(\mathrm{dom}\, f)$ with $\mathbf{x}_k$ converging to a boundary point of $\mathrm{dom}\, f$, we have $\|\nabla f(\mathbf{x}_k)\| \to \infty$.

(ii) f is strictly convex on every convex subset of $\mathrm{dom}(\partial f)$.

If f is Legendre function, its Fenchel conjugate reduces to the Legendre transform, defined by

$$f^*(\mathbf{y}) = \mathbf{x}(\mathbf{y})^\top \mathbf{y} - f(\mathbf{x}(\mathbf{y})),$$

where $\mathbf{x}(\mathbf{y}) = \arg\min_{\mathbf{x}} (\mathbf{x}^\top \mathbf{y} - f(\mathbf{x}))$ is the unique solution to the first-order optimality condition $\nabla f(\mathbf{x}) = \mathbf{y}$.

Examples

Example 1.11 (**Conjugate of the Quadratic Function.**). *Let* $f(\mathbf{x}) = \frac{1}{2}\|\mathbf{x}\|^2$. *Then:*

$$f^*(\mathbf{y}) = \sup_{\mathbf{x}} \left\{ \mathbf{x}^\top \mathbf{y} - \frac{1}{2}\|\mathbf{x}\|_2^2 \right\} = \frac{1}{2}\|\mathbf{y}\|_2^2.$$

Example 1.12 (**Conjugate of the Squared Hinge.**). *Let* $f(x) = \max(x, 0)^2$. *Then:*

$$f^*(y) = \sup_x xy - \max(x, 0)^2 = \begin{cases} \frac{y^2}{4}, & y \geq 0 \\ \infty, & y < 0 \end{cases}.$$

The Legendre transform is not defined in this case since f *is not strictly convex.*

Example 1.13. *Log-sum-exp and negative entropy are conjugates of each other. Please refer to the Example 1.16.*

1.4 Convex Optimization

A standard optimization problem is defined by:

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^d} \quad & f_0(\mathbf{x}) \\ s.t. \quad & f_i(\mathbf{x}) \leq 0, i = 1, \ldots, m \\ & h_j(\mathbf{x}) = 0, j = 1, \ldots, n. \end{aligned} \tag{1.7}$$

Definition 1.9 A standard optimization problem (1.7) is a convex optimization problem if $f_i(\mathbf{x})$ is convex for $i = 0, \ldots, m$ and $h_j(\mathbf{x}) = \mathbf{a}_j^\top \mathbf{x} + b_j$ is an affine function for $j = 1, \ldots, n$.

The problem (1.7) is feasible if there exists at least one point such that all constraints are satisfied, and infeasible otherwise. The set of all feasible points is called the feasible set, denoted by

$$\mathcal{X} = \{\mathbf{x} : f_i(\mathbf{x}) \leq 0, i = 1, \ldots, m, h_i(\mathbf{x}) = 0, j = 1, \ldots, n\}.$$

The Optimal value and optimal solutions

The optimal value of (1.7) is defined as

$$f_* = \inf\{f_0(\mathbf{x}) | \mathbf{x} \in \mathcal{X}\}.$$

where inf returns the greatest value that is less than or equal to all possible objective values at feasible points if such a value exists. For example inf $e^{-x} = 0$. If the problem is infeasible, we let $f_* = \infty$.

A solution $\mathbf{x}_*$ is an optimal solution if it is feasible, i.e., satisfying all constraints, and $f_0(\mathbf{x}_*) = f_*$. Hence, we may have a set of optimal solutions:

$$\mathcal{X}_* = \arg\min\{f_0(\mathbf{x})|\mathbf{x} \in \mathcal{X}\} = \{\mathbf{x} : \mathbf{x} \in \mathcal{X}, f_0(\mathbf{x}) = f_*\}.$$

The optimal solution is unique if the objective is strongly convex.

1.4.1 Local Minima and Global Minima

A solution $\mathbf{x}$ is called a local minima if there is an $R > 0$ such that

$$f_0(\mathbf{x}) = \inf\{f_0(\mathbf{y})|\mathbf{y} \in \mathcal{X}, \|\mathbf{y} - \mathbf{x}\|_2 \leq R\}. \tag{1.8}$$

Theorem 1.1 *For a convex optimization problem, a local minima* $\mathbf{x}$ *is also a global minima.*

Proof. Suppose $\mathbf{x}$ is not a global minima. It means that there exists a feasible $\mathbf{z}$ such that $f_0(\mathbf{z}) < f_0(\mathbf{x})$. Then $\|\mathbf{z} - \mathbf{x}\|_2 > R$ because $\mathbf{x}$ is an optimal solution in the local region $\Omega = \{\mathbf{y} : \|\mathbf{y} - \mathbf{x}\|_2 \leq R\}$.

Let us derive a contradiction. Let $\mathbf{y} = \mathbf{x} + \theta(\mathbf{z} - \mathbf{x})$, where $\theta = \frac{R}{\|\mathbf{x}-\mathbf{z}\|_2}$ such that $\|\mathbf{y} - \mathbf{x}\|_2 \leq \theta\|\mathbf{z} - \mathbf{x}\|_2 \leq R$. Then $f_0(\mathbf{y}) \leq \theta f_0(\mathbf{z}) + (1 - \theta) f_0(\mathbf{x}) < f_0(\mathbf{x})$, which contradicts to the fact that $\mathbf{x}$ is an optimal solution in the region $\Omega = \{\mathbf{y} : \|\mathbf{y} - \mathbf{x}\|_2 \leq R\}$. Hence such an $\mathbf{z}$ does not exist. □

1.4.2 Optimality Conditions

Let us consider a differential objective function f_0.

Theorem 1.2 *For a convex optimization problem (1.7) with non-empty* $\mathcal{X}_*$, $\mathbf{x}$ *is optimal if and only if* $\mathbf{x} \in \mathcal{X}$ *and*

$$\nabla f_0(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) \geq 0, \forall \mathbf{y} \in \mathcal{X}. \tag{1.9}$$

For non-differential function, the above condition is replaced by $\exists \mathbf{v} \in \partial f_0(\mathbf{x})$ such that $\mathbf{v}^\top (\mathbf{y} - \mathbf{x}) \geq 0, \forall \mathbf{y} \in \mathcal{X}$.

Proof. To prove the sufficient condition, we use the convexity of f_0. For any $\mathbf{y} \in \mathcal{X}$, we have

$$f_0(\mathbf{y}) \geq f_0(\mathbf{x}) + \nabla f_0(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) \geq f_0(\mathbf{x}).$$

Hence $\mathbf{x}$ is an optimal solution. Let us prove the necessary condition. If (1.9) does not hold for an $\mathbf{y}$, i.e., $\nabla f_0(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) < 0$, let us consider $\mathbf{z}(t)) = t\mathbf{y} + (1-t)\mathbf{x}$, which is feasible. Thence $\nabla_t f_0(\mathbf{z}(t))|_{t=0} = \nabla f_0(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) < 0$, which means there exists a small $t > 0$ such that $f_0(\mathbf{z}(t)) \leq f_0(\mathbf{z}(0)) = f_0(\mathbf{x})$, which is impossible as $\mathbf{x}$ is an optimal solution. □

When the problem is unconstrained such that $\mathcal{X} = \mathbb{R}^d$, then the optimality condition (1.9) implies that $\mathbf{x}$ is optimal if and only if $\nabla f_0(\mathbf{x}) = 0$.

Lemma 1.2 *For a convex optimization problem (1.7), if f_0 is strongly convex, then $\mathcal{X}_*$ contains only a single element if it is not empty.*

Proof. Assume $\mathcal{X}_*$ contains two different solutions $\mathbf{x}_1 \neq \mathbf{x}_2$ such that $f_0(\mathbf{x}_1) = f_0(\mathbf{x}_2)$. We will derive a contradiction. Since f_0 is strongly convex, we have

$$f_0(\mathbf{x}_1) \geq f_0(\mathbf{x}_2) + \partial f_0(\mathbf{x}_2)^\top (\mathbf{x}_1 - \mathbf{x}_2) + \frac{1}{2}\|\mathbf{x}_1 - \mathbf{x}_2\|_2^2.$$

Due to the optimality condition, $\partial f_0(\mathbf{x}_2)^\top (\mathbf{x}_1 - \mathbf{x}_2) \geq 0$, hence $f_0(\mathbf{x}_1) \geq f_0(\mathbf{x}_2) + \frac{1}{2}\|\mathbf{x}_1 - \mathbf{x}_2\|_2^2 > f_0(\mathbf{x}_2)$, which contradicts to the fact $f_0(\mathbf{x}_1) = f_0(\mathbf{x}_2)$.

□

1.4.3 Karush–Kuhn–Tucker (KKT) Conditions

Constrained optimization problems such as (1.7) are often challenging to analyze and solve directly. The Karush-Kuhn-Tucker (KKT) conditions, derived from Lagrangian duality theory, offer first-order necessary conditions for optimality. These conditions can simplify the original problem, sometimes enabling a transformation into a more tractable form or even leading to a closed-form solution.

The Lagrangian function and the Lagrangian dual function

For the constrained optimization (1.7), the Lagrangian function is defined as:

$$L(\mathbf{x}, \lambda, \mu) = f_0(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i f_i(\mathbf{x}) + \sum_{j=1}^{n} \nu_j h_j(\mathbf{x}),$$

where $\lambda_1, \ldots, \lambda_m, \nu_1, \ldots, \nu_n$ are called the Lagrangian multipliers.

The Lagrangian dual function is defined as:

$$g(\lambda, \nu) = \inf_{\mathbf{x}} L(\mathbf{x}, \lambda, \mu).$$

Based on this, we define the Lagrangian dual problem:

$$g_* = \sup_{\lambda \geq 0} g(\lambda, \nu).$$

Regarding the original optimal value f_* and the dual optimal value g_*, we have the following weak duality.

Lemma 1.3 *We always have $g_* \leq f_*$.*

Proof. Let $\mathbf{x}_*$ be an optimal solution to (1.7). For any $\lambda \geq 0, \nu$, we have

$$\begin{aligned} g(\lambda, \nu) &= \inf_{\mathbf{x}} L(\mathbf{x}, \lambda, \mu) \leq L(\mathbf{x}_*, \lambda, \mu) \\ &= f_0(\mathbf{x}_*) + \sum_{i=1}^{m} \lambda_i f_i(\mathbf{x}_*) + \sum_{j=1}^{n} \nu_j h_j(\mathbf{x}_*) \leq f_0(\mathbf{x}_*), \end{aligned}$$

where the last inequality uses the fact $h_j(\mathbf{x}_*) = 0$, $f_i(\mathbf{x}_*) \leq 0$, and $\lambda \geq 0$. The conclusion follows. □

KKT conditions

An interesting scenario is the strong duality where $g_* = f_*$. In such case, we can derive two conditions.

Lemma 1.4 *Suppose that the primal and dual optimal values are attained and equal. Let $\mathbf{x}_*$ be an optimal primal solution and λ_*, ν_* be optimal dual solutions. Assume that f, g_i, h_j are continuously differentiable, then the following conditions hold:*

$$\nabla f_0(\mathbf{x}_*) + \sum_{i=1}^{m} \lambda_{*,i} \nabla f_i(\mathbf{x}_*) + \sum_{j=1}^{n} \nu_{*,j} \nabla h_j(\mathbf{x}_*) = 0, \tag{1.10}$$

$$\lambda_{*,i} f_i(\mathbf{x}_*) = 0, i = 1, \ldots, m, \tag{1.11}$$

where the second condition is called the complementary slackness.

Proof. First, we have

$$\begin{aligned} g_* &= \sup_{\lambda \geq 0} g(\lambda, \nu) = g(\lambda_*, \nu_*) = \inf_{\mathbf{x}} L(\mathbf{x}, \lambda_*, \nu_*) \\ &= \inf_{\mathbf{x}} f_0(\mathbf{x}) + \sum_{i=1}^{m} \lambda_{*,i} f_i(\mathbf{x}) + \sum_{j=1}^{n} \nu_{*,j} h_j(\mathbf{x}) \\ &\leq f_0(\mathbf{x}_*) + \sum_{i=1}^{m} \lambda_{*,i} f_i(\mathbf{x}_*) + \sum_{j=1}^{n} \nu_{*,j} h_j(\mathbf{x}_*) \\ &\leq f_0(\mathbf{x}_*) = f_*. \end{aligned}$$

Since $g_* = f_*$, the inequalities will become equalities. The first equality is

$$\inf_{\mathbf{x}} f_0(\mathbf{x}) + \sum_{i=1}^{m} \lambda_{*,i} f_i(\mathbf{x}) + \sum_{j=1}^{n} \nu_{*,j} h_j(\mathbf{x}) = f_0(\mathbf{x}_*) + \sum_{i=1}^{m} \lambda_{*,i} f_i(\mathbf{x}_*) + \sum_{j=1}^{n} \nu_{*,j} h_j(\mathbf{x}_*),$$

which implies that $\mathbf{x}_*$ optimizes $L(\mathbf{x}, \lambda_*, \nu_*)$. Hence, by the first-order optimality condition, we have $\nabla_{\mathbf{x}} L(\mathbf{x}, \lambda_*, \nu_*) = 0$, which is (1.10). The second equality is

$$f_0(\mathbf{x}_*) + \sum_{i=1}^{m} \lambda_{*,i} f_i(\mathbf{x}_*) + \sum_{j=1}^{n} \nu_{*,j} h_j(\mathbf{x}_*) = f_0(\mathbf{x}_*),$$

which implies $\lambda_{*,i} f_i(\mathbf{x}_*) = 0, \forall, i$ because $\lambda_{*,i} f_i(\mathbf{x}_*) \leq 0, \forall, i$ and they cannot be larger than zero; otherwise the equality will not hold. □

KKT conditions

Assume that f, g_i, h_j are continuously differentiable. Let $\mathbf{x}_*$ be an optimal primal solution and λ_*, ν_* be optimal dual solutions. The KKT conditions are:

$$\text{(Stationarity)} \quad \nabla f_0(\mathbf{x}_*) + \sum_{i=1}^{m} \lambda_{*,i} \nabla f_i(\mathbf{x}_*) + \sum_{j=1}^{n} \nu_{*,j} \nabla h_j(\mathbf{x}_*) = 0,$$

$$\text{(Primal feasibility)} \quad f_i(\mathbf{x}_*) \leq 0, \quad h_j(\mathbf{x}_*) = 0, \forall i, j,$$

$$\text{(Dual feasibility)} \quad \lambda_{*,i} \geq 0, \forall i,$$

$$\text{(Complementary slackness)} \quad \lambda_{*,i} f_i(\mathbf{x}_*) = 0, \forall i.$$

Slater's condition

How to ensure the strong duality holds? Constraint qualifications have been developed as sufficient conditions of strong duality. One simple constraint qualification is Slater's condition for a convex optimization problem: There exists an $\mathbf{x} \in \text{relint}(D)$ (where relint denotes the relative interior of the convex set $D := \cap_{i=0}^{m} \text{dom}(f_i)$ such that

$$f_i(\mathbf{x}) < 0, \forall i, \quad \text{and} \quad \mathbf{a}_j^\top \mathbf{x} + b_j = 0, \forall j.$$

An important theorem of Lagrangian duality is that the strong duality holds when **the primal problem is convex** and Slater's condition holds. This suggests a tangible approach to compute $\mathbf{x}_*$ or transform the original problem into a simplified one. First, we solve the dual problem to obtain an optimal dual solution (λ_*, ν_*):

$$(\lambda_*, \nu_*) = \arg\max_{\lambda \geq 0, \nu} g(\lambda, \nu). \tag{1.12}$$

Then we use the stationarity condition of KKT conditions to derive a close form of $\mathbf{x}_*$. In addition, we have

$$\min_{\mathbf{x}}\{f_0(\mathbf{x}),\ s.t.\ \mathbf{x}\in\mathcal{X}\} = \max_{\lambda\geq 0,\nu} g(\lambda,\nu).$$

Examples

Example 1.14 (Dual of Distributionally Robust optimization (DRO)). *The following problem often arises in robust machine learning:*

$$f(\ell_1,\ldots,\ell_n) = \max_{\mathbf{p}\in\Delta_n} \sum_{i=1}^n p_i\ell_i - \tau\sum_{i=1}^n q_i\phi(p_i/q_i),$$

where $\tau\geq 0, \mathbf{q}\in\Delta_n$ and $\phi(t):\mathbb{R}^+\to\mathbb{R}$ is a proper closed convex function and has a minimum value zero that is attained at $t=1$. Let us derive its dual problem. We write the above problem as a standard convex optimization problem:

$$\begin{aligned}&\min_{\mathbf{p}} -\sum_{i=1}^n p_i\ell_i + \tau\sum_i q_i\phi(p_i/q_i)\\ &s.t.\ \sum_{i=1}^n p_i = 1.\end{aligned}$$

where the constraint $p_i\geq 0$ is enforced by the domain of $\phi(t)$.
We define the Lagrangian function:

$$L(\mathbf{p},\nu) = -\sum_{i=1}^n p_i\ell_i + \tau\sum_{i=1}^n q_i\phi(p_i/q_i) + \nu\left(\sum_{i=1}^n p_i - 1\right).$$

Let us define

$$\phi^*(s) = \max_{t\geq 0} ts - \phi(t). \tag{1.13}$$

By minimizing over $\mathbf{p}\geq 0$, we have

$$\begin{aligned}g(\nu) &= \min_{\mathbf{p}\geq 0} -\sum_{i=1}^n p_i(\ell_i-\nu) + \tau\sum_{i=1}^n q_i\phi(p_i/q_i) - \nu\\ &= -\{\max_{\mathbf{p}\geq 0}\sum_{i=1}^n p_i(\ell_i-\nu) - \tau\sum_{i=1}^n q_i\phi(p_i/q_i)\} - \nu.\end{aligned}$$

With a variable change $\tilde{p} = p/q$, we have

$$g(\nu) = -\max_{\tilde{p}\geq 0} \sum_{i=1}^{n} q_i\{\tilde{p}_i(\ell_i - \nu) - \tau\phi(\tilde{p}_i)\} - \nu$$
$$= -\sum_{i=1}^{n} q_i\{\max_{\tilde{p}_i\geq 0} \tilde{p}_i(\ell_i - \nu) - \tau\phi(\tilde{p}_i)\} - \nu = -\sum_{i=1}^{n} \tau q_i \phi^*\left(\frac{\ell_i - \nu}{\tau}\right) - \nu.$$

Since the Slater's condition holds ($p_i = 1/n$ satisfies), we have

$$\min_{\mathbf{p}\in\Delta} -\sum_{i=1}^{n} p_i\ell_i + \tau\sum_{i=1}^{n} q_i\phi(p_i/q_i)$$
$$= \max_{\nu} g(\nu) = -\left\{\min_{\nu} \sum_{i=1}^{n} \tau q_i \phi^*\left(\frac{\ell_i - \nu}{\tau}\right) + \nu\right\}.$$

Hence,

$$\max_{\mathbf{p}\in\Delta} \sum_{i=1}^{n} p_i\ell_i - \tau\sum_{i=1}^{n} q_i\phi(p_i/q_i) = \min_{\nu} \sum_{i=1}^{n} \tau q_i \phi^*\left(\frac{\ell_i - \nu}{\tau}\right) + \nu. \quad (1.14)$$

Example 1.15 (**Conjugate of ϕ functions.**). *We can derive ϕ^* for three cases below (exercise):*

- $\phi(t) = (t-1)^2$*:*

$$\phi^*(y) = \max_{t\geq 0} yt - (t-1)^2 = \begin{cases} \frac{1}{4}y^2 + y & \text{if } y \geq -2 \\ -1 & \text{o.w.} \end{cases}$$

- $\phi(t) = t\log t - t + 1$ *and*

$$\phi^*(y) = \max_{t\geq 0} yt - (t\log t - t + 1) = \exp(y) - 1.$$

- $\phi(t) = \mathbb{I}_{0-\infty}(t \leq 1/\alpha)$ *for* $\alpha \in (0, 1]$*:*

$$\phi^*(y) = \max_{t\geq 0} yt - \mathbb{I}_{0-\infty}(t \leq 1/\alpha) = \frac{[y]_+}{\alpha}.$$

Example 1.16 (**KKT conditions of DRO with a KL divergence**). *Let us consider a special case of Example 1.14 with $\phi(t) = t\log t - t + 1$:*

$$f(\ell_1, \ldots, \ell_n) = \max_{\mathbf{p}\in\Delta_n} \sum_{i=1}^{n} p_i\ell_i - \tau\sum_{i=1}^{n} p_i \log\frac{p_i}{q_i}. \quad (1.15)$$

We can derive the following KKT conditions:

$$(\ell_i - \nu_*) - \tau(\log \frac{p_i^*}{q_i} + 1) = 0, \forall i \Rightarrow p_i^* = q_i \exp\left(\frac{\ell_i - \nu_* - \tau}{\tau}\right),$$

$$\sum_{i=1}^{n} p_i^* = 1.$$

As a result, we can derive

$$p_i^* = \frac{q_i \exp(\frac{\ell_i}{\tau})}{\sum_{i=1}^n q_i \exp(\frac{\ell_i}{\tau})} \tag{1.16}$$

$$f(\ell_1, \ldots, \ell_n) = \tau \log\left(\sum_{i=1}^{n} q_i \exp\left(\frac{\ell_i}{\tau}\right)\right). \tag{1.17}$$

1.5 Basic Lemmas

Below, we present some basic lemmas that are useful for the presentation and analysis in later chapters.

Lemma 1.5 *For a L-smooth convex function w.r.t. $\|\cdot\|_2$, the following conditions are equivalent:*

(a) $0 \leq f(\mathbf{y}) - f(\mathbf{x}) - \nabla f(\mathbf{x})^\top(\mathbf{y} - \mathbf{x}) \leq \frac{L}{2}\|\mathbf{x} - \mathbf{y}\|_2^2$;
(b) $\frac{1}{2L}\|\nabla f(\mathbf{y}) - \nabla f(\mathbf{x})\|_2^2 \leq f(\mathbf{y}) - f(\mathbf{x}) - \nabla f(\mathbf{x})^\top(\mathbf{y} - \mathbf{x})$;
(c) $\frac{1}{L}\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|_2^2 \leq (\nabla f(\mathbf{x}) - \nabla f(\mathbf{y}))^\top(\mathbf{x} - \mathbf{y}) \leq L\|\mathbf{x} - \mathbf{y}\|_2^2$;
(d) $\frac{\alpha(1-\alpha)}{2L}\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|_2^2 \leq \alpha f(\mathbf{x}) + (1-\alpha)f(\mathbf{y}) - f(\alpha\mathbf{x} + (1-\alpha)\mathbf{y}) \leq \alpha(1-\alpha)\frac{L}{2}\|\mathbf{x} - \mathbf{y}\|_2^2$.

Proof. Let us prove (a). Since $\frac{df(\mathbf{x}+\gamma\mathbf{p})}{d\gamma} = \nabla f(\mathbf{x}+\gamma\mathbf{p})^\top\mathbf{p}$, according to *Taylor Theory*

$$f(\mathbf{x} + \mathbf{p}) = f(\mathbf{x}) + \int_0^1 \nabla f(\mathbf{x} + \gamma\mathbf{p})^\top\mathbf{p} d\gamma$$

Let $\mathbf{y} = \mathbf{x} + \mathbf{p}$:

$$
\begin{aligned}
&f(\mathbf{y}) - f(\mathbf{x}) - \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) \\
&\quad = \int_0^1 \nabla f(\mathbf{x} + \gamma(\mathbf{y} - \mathbf{x}))^\top (\mathbf{y} - \mathbf{x}) d\gamma - \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) \\
&\quad = \int_0^1 \nabla f(\mathbf{x} + \gamma(\mathbf{y} - \mathbf{x}))^\top (\mathbf{y} - \mathbf{x}) - \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) d\gamma \\
&\quad \leq \int_0^1 \|\nabla f(\mathbf{x} + \gamma(\mathbf{y} - \mathbf{x})) - \nabla f(\mathbf{x})\|_2 \|\mathbf{p}\|_2 d\gamma \\
&\quad \leq \int_0^1 L\|\gamma \mathbf{p}\|_2 \|\mathbf{p}\|_2 d\gamma = \frac{L}{2}\|\mathbf{y} - \mathbf{x}\|_2^2.
\end{aligned}
$$

Let us prove (b). Define $\phi(\mathbf{z}) = f(\mathbf{z}) - \nabla f(\mathbf{x})^\top \mathbf{z}$. We can conclude that $\mathbf{z}^* = \mathbf{x}$ (by the first-order optimality) and that $\phi(\mathbf{z})$ is also convex & L-smooth if f is convex & L-smooth.

$$
\begin{aligned}
\phi(\mathbf{x}) = \min_{\mathbf{z}} \phi(\mathbf{z}) &\leq \min_{\mathbf{z}} \left\{ \phi(\mathbf{y}) + \nabla \phi(\mathbf{y})^\top (\mathbf{z} - \mathbf{y}) + \frac{L}{2}\|\mathbf{z} - \mathbf{y}\|_2^2 \right\} \\
&\overset{r = \mathbf{z} - \mathbf{y}}{=} \min_r \left\{ \phi(\mathbf{y}) + \nabla \phi(\mathbf{y})^\top r + \frac{L}{2}\|r\|_2^2 \right\} \\
&\overset{\text{solve } r}{=} \phi(\mathbf{y}) - \frac{\|\phi(\mathbf{y})\|_2^2}{L} + \frac{\|\nabla \phi(\mathbf{y})\|_2^2}{2L} = \phi(\mathbf{y}) - \frac{\|\nabla \phi(\mathbf{y})\|_2^2}{2L}.
\end{aligned}
$$

Then, we have $2L(\phi(\mathbf{y}) - \phi(\mathbf{x})) \geq \|\nabla \phi(\mathbf{y})\|_2^2$, which prove the result by plugging in $\phi(\mathbf{z}) = f(\mathbf{z}) - \nabla f(\mathbf{x})^\top \mathbf{z}$ and $\nabla \phi(\mathbf{z}) = \nabla f(\mathbf{z}) - \nabla f(\mathbf{x})$.

Let us prove (c). According to part (b) we have

$$
\frac{1}{2L}\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|_2^2 \leq f(\mathbf{y}) - f(\mathbf{x}) - \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}).
$$

Similarly,

$$
\frac{1}{2L}\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|_2^2 \leq f(\mathbf{x}) - f(\mathbf{y}) - \nabla f(\mathbf{y})^\top (\mathbf{x} - \mathbf{y}).
$$

Summing up the above two inequalities leads to

$$
\frac{1}{L}\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|_2^2 \leq (\nabla f(\mathbf{x}) - \nabla f(\mathbf{y}))^\top (\mathbf{x} - \mathbf{y}).
$$

Let us prove (d). Let $\mathbf{x}_\alpha = \alpha \mathbf{x} + (1 - \alpha)\mathbf{y}$. From (a) and (b), we have

$$\frac{1}{2L}\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{x}_\alpha)\|_2^2 \le f(\mathbf{x}) - (f(\mathbf{x}_\alpha) + \nabla f(\mathbf{x}_\alpha)^\top (1-\alpha)(\mathbf{x}-\mathbf{y}))$$
$$\le \frac{L}{2}\|(1-\alpha)(\mathbf{x}-\mathbf{y})\|_2^2,$$
$$\frac{1}{2L}\|\nabla f(\mathbf{y}) - \nabla f(\mathbf{x}_\alpha)\|_2^2 \le f(\mathbf{y}) - (f(\mathbf{x}_\alpha) + \nabla f(\mathbf{x}_\alpha)^\top \alpha(\mathbf{y}-\mathbf{x}))$$
$$\le \frac{L}{2}\|\alpha(\mathbf{y}-\mathbf{x})\|_2^2.$$

Multiplying the first by α and the second by $1-\alpha$, we can prove part (d), where the lower bound is as

$$\frac{\alpha}{2L}\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{x}_\alpha)\|_2^2 + \frac{1-\alpha}{2L}\|\nabla f(\mathbf{y}) - \nabla f(\mathbf{x}_\alpha)\|_2^2 \ge \frac{\alpha(1-\alpha)}{2L}\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|_2^2$$

by applying the Young's inequality $\|\mathbf{a}-\mathbf{b}\|_2^2 \le (1+\beta)\|\mathbf{a}-\mathbf{c}\|_2^2 + (1+\frac{1}{\beta})\|\mathbf{b}-\mathbf{c}\|_2^2$ with $\beta = \alpha/(1-\alpha)$. □

Lemma 1.6 *If f is differentiable and μ-strongly convex w.r.t $\|\cdot\|_2$, the following conditions are equivalent:*

(a) $f(\mathbf{y}) - f(\mathbf{x}) - \nabla f(\mathbf{x})^\top(\mathbf{y}-\mathbf{x}) \ge \frac{\mu}{2}\|\mathbf{x}-\mathbf{y}\|_2^2$;
(b) $f(\mathbf{y}) - f(\mathbf{x}) - \nabla f(\mathbf{x})^\top(\mathbf{y}-\mathbf{x}) \le \frac{1}{2\mu}\|\nabla f(\mathbf{y}) - \nabla f(\mathbf{x})\|_2^2$;
(c) $\mu\|\mathbf{x}-\mathbf{y}\|_2^2 \le (\nabla f(\mathbf{x}) - \nabla f(\mathbf{y}))^\top(\mathbf{x}-\mathbf{y}) \le \frac{1}{\mu}\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|_2^2$;
(d) $\frac{\alpha(1-\alpha)\mu}{2}\|\mathbf{x}-\mathbf{y}\|_2 \le \alpha f(\mathbf{x}) + (1-\alpha)f(\mathbf{y}) - f(\alpha\mathbf{x}+(1-\alpha)\mathbf{y}) \le \alpha(1-\alpha)\frac{1}{2\mu}\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|_2^2$.

From (a) we can derive an useful inequality for strongly convex optimization $\mathbf{x}_* = \arg\min_{\mathbf{x}} f(\mathbf{x})$, i.e., for any $\mathbf{x}$, we have

$$f(\mathbf{x}) \ge f(\mathbf{x}_*) + \nabla f(\mathbf{x}_*)^\top(\mathbf{x}-\mathbf{x}_*) + \frac{\mu}{2}\|\mathbf{x}-\mathbf{x}_*\|_2^2 \ge f(\mathbf{x}_*) + \frac{\mu}{2}\|\mathbf{x}-\mathbf{x}_*\|_2^2. \quad (1.18)$$

Proof of (b). Define $\phi(\mathbf{z}) = f(\mathbf{z}) - \nabla f(\mathbf{x})^\top \mathbf{z}$. We can conclude that $\mathbf{z}^* = \mathbf{x}$ (by the first-order optimality) and that $\phi(\mathbf{z})$ is also convex & μ-strongly convex since f is convex & μ-strongly convex.

$$\phi(\mathbf{x}) = \min_{\mathbf{z}} \phi(\mathbf{z}) \ge \min_{\mathbf{z}} \left\{\phi(\mathbf{y}) + \nabla\phi(\mathbf{y})^\top(\mathbf{z}-\mathbf{y}) + \frac{\mu}{2}\|\mathbf{z}-\mathbf{y}\|_2^2\right\}$$
$$\overset{r=\mathbf{z}-\mathbf{y}}{=} \min_r \left\{\phi(\mathbf{y}) + \nabla\phi(\mathbf{y})^\top r + \frac{\mu}{2}\|r\|_2^2\right\}$$
$$\overset{\text{solve } r}{=} \phi(\mathbf{y}) - \frac{\|\phi(\mathbf{y})\|_2^2}{2\mu}.$$

Then, we have $2\mu(\phi(\mathbf{y}) - \phi(\mathbf{x})) \le \|\phi(\mathbf{y})\|_2^2$, which prove the result by plugging in $\phi(\mathbf{z}) = f(\mathbf{z}) - \nabla f(\mathbf{x})^\top \mathbf{z}$ and $\nabla\phi(\mathbf{z}) = \nabla f(\mathbf{z}) - \nabla f(\mathbf{x})$.

part (b), (c), (d) can be proved similarly as the previous lemma. □

Lemma 1.7 *If $r(\cdot)$ is μ-strongly convex w.r.t $\|\cdot\|_2$ and*

$$\text{prox}_{\eta r}(\mathbf{z}_1) := \arg\min_{\mathbf{w}} r(\mathbf{w}) + \frac{1}{2\eta}\|\mathbf{w}-\mathbf{z}_1\|_2^2, \tag{1.19}$$

$$\text{prox}_{\eta r}(\mathbf{z}_2) := \arg\min_{\mathbf{w}} r(\mathbf{w}) + \frac{1}{2\eta}\|\mathbf{w}-\mathbf{z}_2\|_2^2\,, \tag{1.20}$$

then we have $\|\text{prox}_{\eta r}(\mathbf{z}_1) - \text{prox}_{\eta r}(\mathbf{z}_2)\|_2 \le \frac{1}{1+\mu\eta}\|\mathbf{z}_1-\mathbf{z}_2\|_2$.

Proof. First, we can see that when $r = 0$, the conclusion trivially holds. Next, we prove it when r is present.

By the optimality of $\text{prox}_{\eta r}(\mathbf{z}_1)$ and $\text{prox}_{\eta r}(\mathbf{z}_1)$ we have

$$\begin{aligned}\mathbf{u} :=& \frac{\mathbf{z}_1 - \text{prox}_{\eta r}(\mathbf{z}_1)}{\eta} \in \partial r(\text{prox}_{\eta r}(\mathbf{z}_1))\\ \mathbf{v} :=& \frac{\mathbf{z}_2 - \text{prox}_{\eta r}(\mathbf{z}_2)}{\eta} \in \partial r(\text{prox}_{\eta r}(\mathbf{z}_2)).\end{aligned}$$

Since $r(\mathbf{x})$ is μ-strongly convex, we have

$$\begin{aligned} r(\text{prox}_{\eta r}(\mathbf{z}_1)) \ge\ & r(\text{prox}_{\eta r}(\mathbf{z}_2)) + \mathbf{v}^\top(\text{prox}_{\eta r}(\mathbf{z}_1) - \text{prox}_{\eta r}(\mathbf{z}_2))\\ &+ \frac{\mu}{2}\|\text{prox}_{\eta r}(\mathbf{z}_1) - \text{prox}_{\eta r}(\mathbf{z}_2)\|_2^2\\ r(\text{prox}_{\eta r}(\mathbf{z}_2)) \ge\ & r(\text{prox}_{\eta r}(\mathbf{z}_1)) + \mathbf{u}^\top(\text{prox}_{\eta r}(\mathbf{z}_2) - \text{prox}_{\eta r}(\mathbf{z}_1))\\ &+ \frac{\mu}{2}\|\text{prox}_{\eta r}(\mathbf{z}_1) - \text{prox}_{\eta r}(\mathbf{z}_2)\|_2^2. \end{aligned}$$

Adding them together, we have

$$\begin{aligned}&\mu\|\text{prox}_{\eta r}(\mathbf{z}_1) - \text{prox}_{\eta r}(\mathbf{z}_2)\|_2^2 \le (\mathbf{u}-\mathbf{v})^\top(\text{prox}_{\eta r}(\mathbf{z}_1) - \text{prox}_{\eta r}(\mathbf{z}_2))\\ &= \frac{1}{\eta}(\mathbf{z}_1 - \mathbf{z}_2 + \text{prox}_{\eta r}(\mathbf{z}_2) - \text{prox}_{\eta r}(\mathbf{z}_1))^\top(\text{prox}_{\eta r}(\mathbf{z}_1) - \text{prox}_{\eta r}(\mathbf{z}_2)),\end{aligned}$$

which implies

$$\begin{aligned}&(\mu + \frac{1}{\eta})\|\text{prox}_{\eta r}(\mathbf{z}_1) - \text{prox}_{\eta r}(\mathbf{z}_2)\|_2^2 \le \frac{1}{\eta}(\mathbf{z}_1-\mathbf{z}_2)^\top(\text{prox}_{\eta r}(\mathbf{z}_1) - \text{prox}_{\eta r}(\mathbf{z}_2))\\ &\le \frac{1}{\eta}\|\mathbf{z}_1-\mathbf{z}_2)\|_2\|\text{prox}_{\eta r}(\mathbf{z}_1) - \text{prox}_{\eta r}(\mathbf{z}_2)\|_2.\end{aligned}$$

Thus $\|\text{prox}_{\eta r}(\mathbf{z}_1) - \text{prox}_{\eta r}(\mathbf{z}_2)\|_2 \le \frac{1}{\mu\eta+1}\|\mathbf{z}_1-\mathbf{z}_2\|_2$. □

Lemma 1.8 *For a proper closed convex function f, the following holds:*

(i) if f is G-Lipchitz continuous w.r.t $\|\cdot\|_2$, then $dom(f^*)$ is bounded and for any $\mathbf{y} \in dom(f^*)$, we have $\|\mathbf{y}\|_2 \le G$;

(ii) if $\mathbf{x}_* \in \arg\max_{\mathbf{x}}\{\mathbf{x}^\top \mathbf{y}_* - f(\mathbf{x})\}$*, then* $\mathbf{y}_* \in \arg\max_{\mathbf{y}}\{\mathbf{y}^\top \mathbf{x}_* - f^*(\mathbf{y})\}$*. Equivalently,* $\mathbf{y}_* \in \partial f(\mathbf{x}_*)$ *(or* $\mathbf{x}_* \in \partial f^*(\mathbf{y}_*)$*);*

(iii) if f *is further a Legendre function, then* $f(\mathbf{x}_*) + f^*(\mathbf{y}_*) = \mathbf{x}_*^\top \mathbf{y}_*$ *if and only if* $\mathbf{y}_* = \nabla f(\mathbf{x}_*)$*, and* $\nabla f^* = (\nabla f)^{-1}$.

Proof. Let us prove (i). For any $\mathbf{y}$ with $\|\mathbf{y}\|_2 > G$, let $\mathbf{u} = \mathbf{y}/\|\mathbf{y}\|_2$ and take $\mathbf{x} = t\mathbf{u}$. By Lipschitz continuity, $f(t\mathbf{u}) \le f(0) + Gt$, hence

$$\mathbf{y}^\top t\mathbf{u} - f(t\mathbf{u}) \ge t(\|\mathbf{y}\|_2 - G) - f(0) \to +\infty,$$

so $f^*(\mathbf{y}) = +\infty$ and thus $\mathbf{y} \notin \mathrm{dom}(f^*)$.

Next, we prove (ii). Since $\mathbf{x}_*$ attains the supremum in the definition of $f^*(\mathbf{y}_*)$, we have $\mathbf{y}_* \in \partial f(\mathbf{x}_*)$ according to the optimality condition and $f^*(\mathbf{y}_*) = \mathbf{x}_*^\top \mathbf{y}_* - f(\mathbf{x}_*)$. Using $f^{**} = f$, we obtain

$$f(\mathbf{x}_*) = \sup_{\mathbf{y}}\{\mathbf{y}^\top \mathbf{x}_* - f^*(\mathbf{y})\} = \mathbf{x}_*^\top \mathbf{y}_* - f^*(\mathbf{y}_*),$$

and the above equality shows that $\mathbf{y}_*$ attains the supremum. Hence, $\mathbf{y}_* \in \arg\max_{\mathbf{y}}\{\mathbf{y}^\top \mathbf{x}_* - f^*(\mathbf{y})\}$, and $\mathbf{x}_* \in \partial f^*(\mathbf{y}_*)$.

Lastly, we prove (iii). By definition, $f^*(\mathbf{y}_*)$ is the supremum of the concave function $F(\mathbf{x}) = \mathbf{y}_*^\top \mathbf{x} - f(\mathbf{x})$. If this supremum is attained at $\mathbf{x}_* \in \mathbb{R}^d$, then $\nabla F(\mathbf{x}_*) = 0$, which is to say $\mathbf{y}_* = \nabla f(\mathbf{x}_*)$. On the other hand, if $\mathbf{y}_* = \nabla f(\mathbf{x}_*)$, then $\mathbf{x}_*$ is a maximizer of $F(\mathbf{x})$, and therefore $f^*(\mathbf{y}_*) = \mathbf{y}_*^\top \mathbf{x}_* - f(\mathbf{x}_*)$. Using the this result twice,

$$\begin{aligned} \mathbf{y} = \nabla f(\mathbf{x}) \quad &\text{if and only if} \quad f(\mathbf{x}) + f^*(\mathbf{y}) = \mathbf{x}^\top \mathbf{y} \\ \mathbf{x} = \nabla f^*(\mathbf{y}) \quad &\text{if and only if} \quad f^*(\mathbf{y}) + f^{**}(\mathbf{x}) = \mathbf{x}^\top \mathbf{y}. \end{aligned}$$

Since $f^** = f$, then $\mathbf{x} = \nabla f^{-1}(\mathbf{y}) = \nabla f^*(\mathbf{y})$. Hence $(\nabla f)^{-1} = \nabla f^*$. □

Lemma 1.9 *If* f *is* μ*-strongly convex w.r.t* $\|\cdot\|_2$*, then its Fenchel conjugate is* $1/\mu$*-smooth. Similarly if* f *is* L*-smooth and convex w.r.t* $\|\cdot\|_2$*, then its Fenchel conjugate is* $1/L$*-strongly convex.*

Proof. Let $f^*(\mathbf{y}) = \max_{\mathbf{x}} \mathbf{x}^\top \mathbf{y} - f(\mathbf{x})$ be the Fenchel conjugate of f.

Suppose f is μ-strongly convex. let $\mathbf{x}(\mathbf{y}) = \arg\max_{\mathbf{x}} \mathbf{x}^\top \mathbf{y} - f(\mathbf{x})$. Then $\nabla f^*(\mathbf{y}) = \mathbf{x}(\mathbf{y})$ due to the Danskin Theorem. Similar to the previous lemma, we can prove that

$$\|\mathbf{x}(\mathbf{y}_1) - \mathbf{x}(\mathbf{y}_2)\|_2 \le \frac{1}{\mu}\|\mathbf{y}_1 - \mathbf{y}_2\|_2,$$

which proves the Lipchitz continuity of $\nabla f^*(\mathbf{y})$ and hence the smoothness of f^*.

Suppose f is L-smooth and convex. Let us prove f^* is $1/L$-strongly convex. Let us consider $\mathbf{y}_1, \mathbf{y}_2$. Let $\mathbf{x}_1 \in \arg\max_{\mathbf{x}} \mathbf{x}^\top \mathbf{y}_1 - f(\mathbf{x})$ and $\mathcal{X}_2 = \arg\max_{\mathbf{x}} \mathbf{x}^\top \mathbf{y}_2 - f(\mathbf{x})$. Then $\nabla f(\mathbf{x}_1) = \mathbf{y}_1$. For any $\mathbf{x}_2 \in \mathcal{X}_2$, we have $\nabla f(\mathbf{x}_2) = \mathbf{y}_2$. Given that

$$f^*(\mathbf{y}_1) = \mathbf{x}_1^\top \mathbf{y}_1 - f(\mathbf{x}_1), \quad f^*(\mathbf{y}_2) = \mathbf{x}_2^\top \mathbf{y}_2 - f(\mathbf{x}_2),$$

then

$$
\begin{aligned}
&f^*(\mathbf{y}_1) - f^*(\mathbf{y}_2) - \mathbf{x}_2^\top(\mathbf{y}_1 - \mathbf{y}_2) \\
&= \mathbf{x}_1^\top \mathbf{y}_1 - f(\mathbf{x}_1) - (\mathbf{x}_2^\top \mathbf{y}_2 - f(\mathbf{x}_2)) - \mathbf{x}_2^\top(\nabla f(\mathbf{x}_1) - \nabla f(\mathbf{x}_2)) \\
&= f(\mathbf{x}_2) - f(\mathbf{x}_1) + \mathbf{x}_1^\top \nabla f(\mathbf{x}_1) - \mathbf{x}_2^\top \nabla f(\mathbf{x}_2) - \mathbf{x}_2^\top(\nabla f(\mathbf{x}_1) - \nabla f(\mathbf{x}_2)) \\
&= f(\mathbf{x}_2) - f(\mathbf{x}_1) + (\mathbf{x}_1 - \mathbf{x}_2)^\top \nabla f(\mathbf{x}_1) \geq \frac{1}{2L}\|\nabla f(\mathbf{x}_1) - \nabla f(\mathbf{x}_2)\|_2^2 = \frac{1}{2L}\|\mathbf{y}_1 - \mathbf{y}_2\|_2^2,
\end{aligned}
$$

where the last inequality is due to part (b) of Lemma 1.5. Hence, we can conclude the proof by noting that $\partial f^*(\mathbf{y}_2) = \mathrm{conv}(\mathcal{X}_2)$ due to the generalized Danskin theorem. □

Lemma 1.10 *For $\mathbf{p} \in \Delta_n$, the negative entropy function $R(\mathbf{p}) = \sum_{i=1}^n p_i \log p_i$ is 1-strongly convex w.r.t to the ℓ_1 norm $\|\cdot\|_1$.*

Proof. For any $\mathbf{x}, \mathbf{y} \in \Delta_n$, let $f(t) = R(\mathbf{y} + t(\mathbf{x} - \mathbf{y}))$. By the second-order Taylor expansion, for some $t \in (0, 1)$, we have

$$
\begin{aligned}
R(\mathbf{x}) = f(1) &= f(0) + f'(0) + \frac{1}{2}f''(t) \\
&= R(\mathbf{y}) + \nabla R(\mathbf{y})^\top(\mathbf{x} - \mathbf{y}) + \frac{1}{2}(\mathbf{x} - \mathbf{y})^\top \nabla^2 R(\mathbf{y} + t(\mathbf{x} - \mathbf{y}))(\mathbf{x} - \mathbf{y}).
\end{aligned}
$$

Hence it suffices to prove that $\mathbf{v}^\top \nabla^2 R(\mathbf{p})\mathbf{v} \geq \|\mathbf{v}\|_1^2$ for any $\mathbf{p} \in \Delta_n$. This can be seen from the following:

$$
\begin{aligned}
\mathbf{v}^\top \nabla^2 R(\mathbf{p})\mathbf{v} = \sum_{i=1}^d v_i^2 p_i^{-1} &= \left[\sum_i v_i^2 p_i^{-1}\right]\left[\sum_i p_i\right] \geq \left[\sum_i (p_i^{-1/2}|v_i|)p_i^{1/2}\right]^2 \\
&= \left[\sum_i |v_i|\right]^2,
\end{aligned}
$$

where the inequality follows by Cauchy inequality.

□

1.6 History and Notes

This chapter has selectively introduced core concepts from convex optimization that are most pertinent to the algorithms and applications discussed in later chapters. While the treatment here is necessarily concise, readers seeking a more comprehensive foundation are encouraged to consult several classic references.

The text by Rockafellar (1970a) provides one of the most comprehensive and authoritative treatments of convex analysis. The textbook by Boyd and Vandenberghe (2004) is an excellent introduction to convex optimization well suited for engineers.

It covers convex sets, convex functions, duality, and optimality conditions in detail, and emphasizes geometric intuition and practical modeling. Many of the definitions and examples in this chapter are inspired by this text. Bertsekas (2009) offers deep insights into convex analysis, duality theory, and constrained optimization from a classical perspective.

The KKT condition is named after three mathematicians, William Karush, Harold W. Kuhn and Albert W. Tucker. It was known due to Kuhn and Tucker, who first published the conditions in 1951 (Kuhn and Tucker, 2014). Later scholars discovered that the necessary conditions for this problem had been stated by Karush in his master's thesis in 1939 (Karush, 1939). The Danskin Theorem originates from the work of Danskin (1967), while its generalized form for subdifferentiable is attributed to Bertsekas (2005).

Nesterov's *Introductory Lectures on Convex Programming* (Nesterov, 2004) provides a more mathematically rigorous treatment, including several key lemmas on smooth and strongly convex functions (Lemma 1.5 and Lemma 1.6) that are presented in this chapter. It is particularly useful for readers interested in complexity analysis and the theoretical underpinnings of first-order methods. The proof of Lemma 1.10 is due to Nemirovski et al. (2009).

Chapter 2
Introduction: Advanced Machine Learning

Abstract This chapter begins with an introduction to the traditional empirical risk minimization (ERM) framework, using standard label prediction tasks to illustrate its three core components: loss functions, optimization algorithms, and generalization analysis. We then explore advanced learning techniques including distributionally robust optimization (DRO) and group DRO that aim to enhance model robustness under distribution shifts. Building on this foundation, we introduce the empirical X-risk minimization (EXM) paradigm and discuss its applications in modern machine learning. Finally, we present the concept of data prediction for discriminative learning in foundation models. The goals of this chapter are threefold: (i) to provide a cohesive view of how discriminative principles inform objective function design; (ii) to highlight the role of optimization tools for objective design and model training; and (iii) to motivate the need for compositional optimization frameworks.

models fade, but principles endure!

Contents

2.1 Empirical Risk Minimization

What is Machine Learning (ML)?

In 1959, Arthur Samuel, a pioneer in the field of ML, defined Machine Learning as the "*field of study that gives computers the ability to learn without being explicitly programmed*".

Nowadays, machine learning has become the foundation of AI. The essence of machine learning is to learn a model by optimizing an objective function on training data, with the goal of achieving strong generalization to unseen data. This relationship is captured by the formula:

$$\textbf{Machine Learning} = \textbf{Objective} \quad + \quad \textbf{Algorithm} \quad + \quad \textbf{Generalization}.$$

Optimization plays a fundamental role in machine learning, as it underpins (1) the formulation of objective functions, (2) the development of optimization algorithms, and (3) the analysis of generalization error of learned models. Below, we will use the traditional label prediction problem to illustrate the three components.

2.1.1 Discriminative Label Prediction

In supervised learning, the primary objective is often to learn a predictive model from a given set of supervised training data. Let us consider a classical label prediction problem. Denote by $(\mathbf{x}, y)$ a data-label pair, where $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^{d_0}$ denotes the input feature vector, and $y \in \mathcal{Y} = \{1, \ldots, K\}$ is the corresponding label. The goal is to learn a predictive model parameterized by $\mathbf{w} \in \mathcal{W} \subseteq \mathbb{R}^d$ (e.g., a deep neural network), which induces a scoring function $h(\mathbf{w}; \cdot) : \mathcal{X} \to \mathbb{R}^K$. Conceptually, the model can be expressed as $h(\mathbf{w}; \mathbf{x}) = W h_0(\mathbf{w}; \mathbf{x})$, where $h_0(\mathbf{w}; \cdot) : \mathcal{X} \to \mathbb{R}^{d_1}$ is the feature extraction component, and $W \in \mathbb{R}^{K \times d_1}$ is the classification head corresponding to the K classes.

A classical framework for learning such a model is the well-known empirical risk minimization (ERM), which minimizes the empirical risk over the training dataset. To this end, a pointwise loss function $\ell(h(\mathbf{w}; \mathbf{x}), y)$ is defined to measure the discrepancy between the model's prediction $h(\mathbf{w}; \mathbf{x})$ and the true label y. Given a training dataset $\mathcal{S} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$, the ERM problem is formulated as:

$$\min_{\mathbf{w} \in \mathcal{W}} \mathcal{R}_{\mathcal{S}}(\mathbf{w}) := \frac{1}{n} \sum_{i=1}^{n} \ell(h(\mathbf{w}; \mathbf{x}_i), y_i). \tag{2.1}$$

2.1.2 Discriminative Loss Functions

A major element of ERM is the design of the loss function. A common strategy of designing a loss function for label prediction is through a discriminative approach. Below, we introduce several discriminative loss functions.

Logistic Loss

A parameterized probabilistic model is defined to represent the probability of a class label for a given data point as

$$\Pr(y|\mathbf{x};\mathbf{w}) = \frac{\exp([h(\mathbf{w};\mathbf{x})]_y)}{\sum_{l=1}^{K}\exp([h(\mathbf{w};\mathbf{x})]_l)}, \tag{2.2}$$

where $[\cdot]_k$ denotes the k-th element of a vector. The associated loss is derived from the negative log-likelihood, resulting in the multi-class logistic loss, also known as the cross-entropy (CE) loss:

$$\ell(h(\mathbf{w};\mathbf{x}), y) = -\log\frac{\exp([h(\mathbf{w};\mathbf{x})]_y)}{\sum_{l=1}^{K}\exp([h(\mathbf{w};\mathbf{x})]_l)}. \tag{2.3}$$

The resulting method by ERM is commonly referred to as multi-class logistic regression. For binary classification, this loss becomes the binary logistic loss $\ell(h(\mathbf{w};\mathbf{x}), y) = \log(1+\exp(-yh(\mathbf{w};\mathbf{x})))$, where $h(\mathbf{w};\cdot) \in \mathbb{R}$ and $y \in \{1,-1\}$.

Max-Margin Loss

The max-margin loss, introduced by Crammer and Singer and commonly referred to as the Crammer-Singer (CS) loss (Crammer and Singer, 2002), is defined as:

$$\ell(h(\mathbf{w};\mathbf{x}), y) = \max\left(0, \max_{k\neq y}\left(c_{k,y} + [h(\mathbf{w};\mathbf{x})]_k - [h(\mathbf{w};\mathbf{x})]_y\right)\right), \tag{2.4}$$

where $c_{k,y} > 0$ is a margin parameter. This loss seeks to ensure that the prediction score for the ground-truth label, $[h(\mathbf{w};\mathbf{x})]_y$, exceeds the scores of other class labels, $[h(\mathbf{w};\mathbf{x})]_k$ for $k \neq y$, by at least the margin $c_{k,y}$. This method is also known as the multi-class support vector machine. For binary classification, it reduces to the standard hinge loss $\ell(h(\mathbf{w};\mathbf{x}), \mathbf{y}) = \max(0, 1 - yh(\mathbf{w};\mathbf{x}))$ for $h(\mathbf{w};\cdot) \in \mathbb{R}$ and $y \in \{1,-1\}$ with a margin 1.

Label Distributionally Robust (LDR) Loss

Both the CS loss and the CE loss have their strengths and limitations. For example, the CS loss with the margin parameters is more flexible in controlling the discrimination between classes, while it is not consistent and non-smooth in terms of the prediction scores. The CE loss is smooth and consistent but lacks robustness to noise in class labels.

Consistency of a surrogate loss function

The consistency measures whether minimizing a surrogate loss with an infinite number of data also minimizes the Bayes error. More formally, a surrogate loss $\ell(h(\mathbf{x}), y)$ is said to be consistent if any sequence of measurable functions $h^{(n)}$ it holds

$$\mathcal{R}(h^{(n)}) \to \inf_{h\in\mathcal{H}} \mathcal{R}(h) \Rightarrow \mathcal{R}_{0-1}(h^{(n)}) \to \inf_{h\in\mathcal{H}} \mathcal{R}_{0-1}(h),$$

where $\mathcal{R}(h) = \mathbb{E}_{\mathbf{x},y}[\ell(h(\mathbf{x}), y)]$ is the expected risk, $\mathcal{R}_{0-1}(h) = \mathbb{E}_{\mathbf{x},y}[\mathbb{I}(y \neq h(\mathbf{x}))]$ is the Bayes error, and $\mathcal{H}$ is the set of any measurable functions.

In fact, the strengths and limitations of both the CE and CS losses can be better understood within a broader family known as the label-distributionally robust (LDR) loss:

$$\ell_\tau(h(\mathbf{w};\mathbf{x}), y) = \max_{\mathbf{p}\in\Delta_K} \sum_{k=1}^{K} p_k\big([h(\mathbf{w};\mathbf{x})]_k - [h(\mathbf{w};\mathbf{x})]_y + c_{k,y}\big) - \tau \sum_{k=1}^{K} p_k \log(p_k K), \tag{2.5}$$

where $\tau > 0$ is a hyperparameter, $c_{y,y} = 0$, $\mathbf{p} \in \mathbb{R}^K$ is referred to as the label distributional weight vector, and $\Delta_K = \{\mathbf{p} \in \mathbb{R}^K : p_k \geq 0, \sum_{k=1}^K p_k = 1\}$ is a simplex.

It is clear that the LDR loss is defined by solving an optimization problem. Indeed, the above optimization problem follows the distributionally robust optimization (DRO) principle, which is widely used at the level of data as discussed in section 2.2. By treating 'label' as a kind of data, we can unify the LDR loss with other losses discussed later in Section 2.4.

A closed-form solution for $\mathbf{p}$ can be derived using the KKT conditions (cf. Example 1.16), making the LDR loss equivalent to:

$$\ell_\tau(h(\mathbf{w};\mathbf{x}), y) = \tau \log\left(\frac{1}{K}\sum_{k=1}^{K} \exp\left(\frac{[h(\mathbf{w};\mathbf{x})]_k + c_{k,y} - [h(\mathbf{w};\mathbf{x})]_y}{\tau}\right)\right). \tag{2.6}$$

From the perspective of DRO, we can define a more general family of LDR losses using different regularization functions on $\mathbf{p}$ and constrained domains Ω:

$$\bar{\ell}_\tau(h(\mathbf{w};\mathbf{x}),y) = \max_{\mathbf{p}\in\Omega} \sum_{k=1}^{K} p_k\big([h(\mathbf{w};\mathbf{x})]_k - [h(\mathbf{w};\mathbf{x})]_y + c_{k,y}\big) - \tau R(\mathbf{p}). \tag{2.7}$$

where $\Omega \subseteq \Delta_K$ and $R(\mathbf{p})$ is a strongly convex regularizer.

Why it matters:

- The LDR loss (2.6) unifies both the CS and CE losses as special cases. Specifically, the CE loss corresponds to the LDR loss when $\tau = 1$ and $c_{k,y} = 0$ for all k, while the CS loss corresponds to the case $\tau = 0$.
  Moreover, the LDR loss encompasses the Label-Distribution-Aware Margin (LDAM) loss (Cao et al., 2019) when $\tau = 1$ and $c_{k,y} = c_y \propto 1/n_y^{1/4}$ for $k \neq y$, where n_y denotes the number of samples in class y:

$$\ell_{\text{LDAM}}(h(\mathbf{w};\mathbf{x}),y) = -\log\left(\frac{\exp\left([h(\mathbf{w};\mathbf{x})]_y - \frac{C}{n_y^{1/4}}\right)}{\exp\left([h(\mathbf{w};\mathbf{x})]_y - \frac{C}{n_y^{1/4}}\right) + \sum_{l\neq y}\exp([h(\mathbf{w};\mathbf{x})]_l)}\right),$$

  where C is a constant. For imbalanced datasets, this assigns larger margins c_y to minority classes, making it more suitable for handling class imbalance.
- The LDR loss provides insights into the strengths and limitations of CE and CS losses. The regularizer $R(\mathbf{p}) = \sum_{k=1}^{K} p_k \log(p_k K)$ is strongly convex in $\mathbf{p}$, which implies smoothness of the loss in terms of prediction scores due to the duality between smoothness and strong convexity (Lemma 1.9). This strong convexity also contributes to the statistical consistency of the loss (Zhu et al., 2023b). In contrast, the CS loss with $\tau = 0$ lacks this property, and hence suffer from non-smoothness and inconsistency.
- The LDR loss framework enables the design of new losses that are robust to label noise. For instance, when $\tau \to \infty$, the LDR loss reduces to:

$$\ell_\infty(\mathbf{w};\mathbf{x},y) = \frac{1}{K}\sum_{k=1}^{K}\big([h(\mathbf{w};\mathbf{x})]_k - [h(\mathbf{w};\mathbf{x})]_y + c_{k,y}\big).$$

  A remarkable property of this loss is its symmetry: $\sum_{y=1}^{K}\ell_\infty(\mathbf{w};\mathbf{x},y)$ is constant. This symmetry serves as a sufficient condition for robustness to uniform label noise (Ghosh et al., 2017). However, by treating all negative labels equally, it may limit the model's ability to focus on hard negative labels and potentially slow down the learning process. In practice, it is better to tune τ if there is label noise.

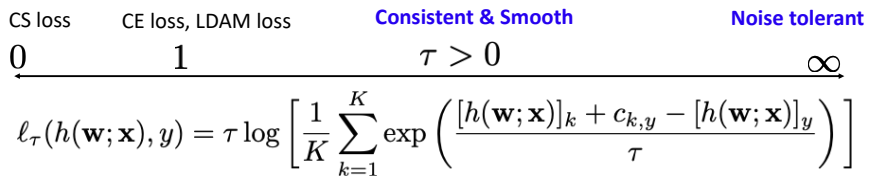


Fig. 2.1: The LDR loss and its special cases by varying τ.

In conclusion, the LDR loss offers flexibility in achieving three desirable properties: max-margin, consistency, and symmetry. In practice, when tuning $\tau \in (0, \infty)$, it may be beneficial to normalize the prediction scores $h(\mathbf{w}; \mathbf{x})$.

Critical: It is worth noting that all the discussed losses are discriminative in nature, aiming to increase the score $[h(\mathbf{w}; \mathbf{x})]_y$ of the true label while decreasing the scores $[h(\mathbf{w}; \mathbf{x})]_k$ of the negative labels ($k \neq y$).

2.1.3 Need of Optimization Algorithms

To address the ERM problem in the context of large-scale data (i.e., a substantial number of training examples), first-order stochastic algorithms are commonly employed. These include stochastic gradient descent (SGD), stochastic momentum methods, and adaptive gradient methods. For instance, the update rule of classical SGD for solving (2.1) with $\mathcal{W} = \mathbb{R}^d$ is given by:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \frac{1}{|\mathcal{B}_t|} \sum_{(\mathbf{x}_i, y_i) \in \mathcal{B}_t} \nabla \ell(h(\mathbf{w}_t; \mathbf{x}_i), y_i), \quad t = 1, \ldots, T, \tag{2.8}$$

where $\eta_t \geq 0$ is the learning rate (or step size), and $\mathcal{B}_t$ denotes a random mini-batch data sampled from the full dataset. The concern of designing an optimization algorithm is how fast the algorithm can converge to a (near) optimal solution. We will discuss the design and analysis of classical stochastic optimization algorithms in Chapter 3.

Critical: A critical assumption in conventional stochastic optimization algorithms such as SGD is that the gradient $\nabla \ell(h(\mathbf{w}; \mathbf{x}_i), y_i)$ of each individual loss, can be easily computed. This assumption will fail for the logistic loss when the number of classes K is gigantic, e.g. millions or even billions. This challenge will be addressed in this book.

2.1.4 Generalization Analysis

To study the generalization of a model learned by solving ERM, we usually consider the expected risk defined as

$$\mathcal{R}(\mathbf{w}) = \mathbb{E}_{\mathbf{x},y\sim\mathbb{P}}[\ell(h(\mathbf{w};\mathbf{x}), y)]. \tag{2.9}$$

Let $\mathbf{w} = \mathcal{A}(\mathcal{S};\zeta)$ denote a learned model by a randomized algorithm $\mathcal{A}$ for solving ERM that depend on random variables ζ. A standard measure of generalization is given by the **excess risk** defined as $\mathcal{R}(\mathbf{w}) - \mathcal{R}(\mathbf{w}_*)$, where $\mathbf{w}_* \in \arg\min_{\mathbf{u}\in\mathcal{W}} \mathcal{R}(\mathbf{u})$. The following lemma decomposes the excess risk into the optimization error and the generalization error.

Lemma 2.1 *For a learned model* $\mathbf{w} = \mathcal{A}(\mathcal{S};\zeta) \in \mathcal{W}$*, we have*

$$\mathcal{R}(\mathbf{w}) - \mathcal{R}(\mathbf{w}_*) \leq 2\underbrace{\sup_{\mathbf{w}\in\mathcal{W}} |\mathcal{R}(\mathbf{w}) - \mathcal{R}_{\mathcal{S}}(\mathbf{w})|}_{\textit{generalization error}} + \underbrace{\mathcal{R}_{\mathcal{S}}(\mathbf{w}) - \min_{\mathbf{u}\in\mathcal{W}} \mathcal{R}_{\mathcal{S}}(\mathbf{u})}_{\textit{optimization error}},$$

and

$$\mathbb{E}_{\mathcal{S},\zeta}[\mathcal{R}(\mathbf{w}) - \mathcal{R}(\mathbf{w}_*)] = \mathbb{E}_{\mathcal{S},\zeta}[\mathcal{R}(\mathbf{w}) - \mathcal{R}_{\mathcal{S}}(\mathbf{w})] + \mathbb{E}_{\mathcal{S},\zeta}[\mathcal{R}_{\mathcal{S}}(\mathbf{w}) - \min_{\mathbf{u}\in\mathcal{W}} \mathcal{R}_{\mathcal{S}}(\mathbf{u})].$$

Proof.

$$\begin{aligned}\mathcal{R}(\mathbf{w}) - \mathcal{R}(\mathbf{w}_*) &= \mathcal{R}(\mathbf{w}) - \mathcal{R}_{\mathcal{S}}(\mathbf{w}) + \mathcal{R}_{\mathcal{S}}(\mathbf{w}) - \min_{\mathbf{u}\in\mathcal{W}} \mathcal{R}_{\mathcal{S}}(\mathbf{u}) + \min_{\mathbf{u}\in\mathcal{W}} \mathcal{R}_{\mathcal{S}}(\mathbf{u}) - \mathcal{R}(\mathbf{w}_*) \\ &\leq \mathcal{R}(\mathbf{w}) - \mathcal{R}_{\mathcal{S}}(\mathbf{w}) + \mathcal{R}_{\mathcal{S}}(\mathbf{w}) - \min_{\mathbf{u}\in\mathcal{W}} \mathcal{R}_{\mathcal{S}}(\mathbf{u}) + \mathcal{R}_{\mathcal{S}}(\mathbf{w}_*) - \mathcal{R}(\mathbf{w}_*).\end{aligned}$$

This proves the first inequality. By taking expectation over $\mathcal{S}, \zeta$ and noting that $\mathbb{E}_{\mathcal{S}}[\mathcal{R}_{\mathcal{S}}(\mathbf{w}_*) - \mathcal{R}(\mathbf{w}_*)] = 0$, we finish the second inequality. □

Why it matters:

The excess risk can be decomposed into two components: the optimization error, given by $\mathcal{R}_{\mathcal{S}}(\mathbf{w}) - \min_{\mathbf{u}\in\mathcal{W}} \mathcal{R}_{\mathcal{S}}(\mathbf{u})$, and the generalization error which captures the difference between the expected risk and the empirical risk. The generalization error $\sup_{\mathbf{w}\in\mathcal{W}} |\mathcal{R}(\mathbf{w}) - \mathcal{R}_{\mathcal{S}}(\mathbf{w})|$ decreases as the training data size $|\mathcal{S}|$ increases. Bounding the (expected) optimization error is a central focus of this book, approached through the analysis of stochastic optimization algorithms. A brief discussion of the literature on generalization error analysis will be provided at the end of this chapter.

2.2 Robust Optimization

In this section, we introduce advanced machine learning methods based on the principle of robust optimization. Robust optimization is a framework designed to address uncertainty in data. It ensures that the solutions perform well even under worst-case scenarios of data within a specified set of uncertainties.

2.2.1 Distributionally Robust Optimization

Minimizing the average empirical risk often fails to yield a robust model in practice. For instance, the resulting model may perform poorly on minority data (e.g., patients with rare diseases) because the optimization predominantly focuses on majority class data.

Critical: Empirical data may not fully represent the underlying data distribution, leading to generalization issues.

To address these challenges, distributionally robust optimization (DRO) has been extensively studied in machine learning as a means to improve robustness and generalization.

The core idea of DRO is to minimize a robust objective defined over the worst-case distribution of data, perturbed from the empirical distribution. Let us define a set of distributional weights, $\mathbf{p} = (p_1, \ldots, p_n) \in \Delta_n$, where $\Delta_n = \{\mathbf{p} \in \mathbb{R}^n : \sum_{i=1}^n p_i = 1,\ p_i \geq 0\}$, with each element p_i associated with a training sample $\mathbf{x}_i$.

Definition 2.1 (ϕ-divergence) Let $\phi(t) : \mathbb{R}^+ \to \mathbb{R}$ is a proper closed convex function and has a minimum value zero that is attained at $t = 1$. The ϕ-divergence is defined as:

$$D_\phi(\mathbf{p} \,\|\, \mathbf{q}) = \sum_{i=1}^{n} q_i \phi(p_i/q_i). \tag{2.10}$$

ϕ-divergence measures the discrepancy between two distributions $\mathbf{p}$ and $\mathbf{q}$ using the function ϕ. We present two common formulations of DRO based on the ϕ-divergence: regularized DRO and constrained DRO. They differ in how to define the uncertainty set of $\mathbf{p}$.

Below, we use the generic notation $\ell(\mathbf{w}; \mathbf{z})$ to denote the loss of a model $\mathbf{w}$ on a random data point $\mathbf{z}$ following a distribution denoted by $\mathbb{P}$. For supervised learning, this specializes to $\ell(\mathbf{w}; \mathbf{z}) = \ell(h(\mathbf{w}; \mathbf{x}), y)$, where $\mathbf{z} = (\mathbf{x}, y)$.

Definition 2.2 (Regularized DRO)

$$\min_{\mathbf{w}} \hat{\mathcal{R}}_{\mathcal{S}}(\mathbf{w}) := \max_{\mathbf{p} \in \Delta_n} \sum_{i=1}^{n} p_i \ell(\mathbf{w}; \mathbf{z}_i) - \tau D_\phi \left(\mathbf{p} \,\|\, \frac{\mathbf{1}}{n} \right). \tag{2.11}$$

Divergence	$\phi(t)$	$\phi^*(s)$	$D_\phi(\mathbf{p} \parallel \mathbf{q})$
KL	$t\log(t) - t + 1$	$\exp(s) - 1$	$\sum_{i=1}^n p_i \log \frac{p_i}{q_i}$
Burg entropy	$-\log t + t - 1$	$-\log(1-s), s < 1$	$\sum_{i=1}^n q_i \log \frac{q_i}{p_i}$
χ^2	$(t-1)^2$	$\begin{cases} \frac{1}{4}s^2 + s & \text{if } s \geq -2 \\ -1 & \text{o.w.} \end{cases}$	$\sum_{i=1}^n q_i (p_i/q_i - 1)^2$
Hellinger distance	$(\sqrt{t}-1)^2$	$\frac{s}{1-s}, s < 1$	$\sum_i (\sqrt{p_i} - \sqrt{q_i})^2$
Variation distance	$\lvert t-1 \rvert$	$\begin{cases} s & \text{if } s \in [-1,1] \\ -1 & \text{if} s < -1 \end{cases}$	$\sum_i \lvert p_i - q_i \rvert$
CVaR	$\mathbb{I}_{0-\infty}(t \leq 1/\alpha)$	$\frac{[s]_+}{\alpha}$	$\begin{cases} 0 & \text{if } p_i \leq q_i/\alpha, \forall i \\ \infty & \text{o.w} \end{cases}$

Table 2.1: Examples of ϕ-divergence

Definition 2.3 (Constrained DRO)

$$\min_{\mathbf{w}} \hat{\mathcal{R}}_{\mathcal{S}}(\mathbf{w}) := \max_{\mathbf{p} \in \Omega} \sum_{i=1}^n p_i \ell(\mathbf{w}; \mathbf{z}_i) \tag{2.12}$$
$$\text{where} \quad \Omega = \left\{ \mathbf{p} | \mathbf{p} \in \Delta_n, D_\phi \left(\mathbf{p} \parallel \frac{\mathbf{1}}{n} \right) \leq \rho \right\}.$$

The regularized DRO uses a regularization on the $\mathbf{p}$ to implicitly define the uncertainty set, and the constrained DRO uses a constraint on $\mathbf{p}$ to explicitly define the uncertainty set.

The maximization over $\mathbf{p}$ in the DRO formulations simulates a worst-case scenario, thereby enhancing the model's robustness. The DRO objective interpolates between the maximal loss and the average loss:

- Without the ϕ-divergence regularization or constraint (i.e., $\tau = 0$ or $\rho = \infty$), the objective simplifies to the maximal loss among all samples, which is particularly beneficial for handling imbalanced data but is sensitive to outliers.
- Conversely, when $\rho = 0$ or $\tau = \infty$, the DRO objective reduces to the standard empirical risk, which is not sensitive to outliers but no suitable for imbalanced data.

In practice, adding a tunable ϕ-divergence regularization or constraint (via tuning τ or ρ) increases the model's robustness.

A list of ϕ-divergence is presented in Table 2.1. Two commonly used ones in machine learning are presented below:

- KL-Divergence: With $\phi(t) = t\log t - t + 1$, the ϕ-divergence becomes the KL divergence:
$$\mathrm{KL}(\mathbf{p}, \mathbf{q}) = D_\phi(\mathbf{p} \parallel \mathbf{q}) = \sum_{i=1}^n p_i \log(p_i/q_i).$$
- Conditional Value-at-Risk (CVaR): With $\phi(t) = \mathbb{I}_{0-\infty}(t \leq 1/\alpha)$, where $\alpha \in (0,1]$ and $\mathbb{I}_{0-\infty}$ is $0-\infty$ indicator function, the divergence becomes $D_\phi(\mathbf{p} \parallel \mathbf{q}) = 0$ if

$p_i \leq q_i/\alpha \, \forall i$, otherwise $D_\phi(\mathbf{p} \,\|\, \mathbf{q}) = \infty$. The resulting DRO formulation is also known as the empirical CVaR-α.

The Dual form of Regularized DRO

Solving the above DRO formulations requires dealing with a high-dimensional variable $\mathbf{p}$ from a simplex, which will incur additional overhead compared with solving ERM when the number of training data is large. The reason is that it requires performing a projection onto the simplex Δ_n or the constrained simplex $\Omega = \{\mathbf{p} \in \Delta_n, D_\phi\left(\mathbf{p} \,\|\, \frac{\mathbf{1}}{n}\right) \leq \rho\}$. To reduce this overhead, one approach is to convert the problem into unconstrained one using the Langrangian dual theory based on the convex conjugate of ϕ function.

Proposition 2.1 (**Dual form of Regularized DRO).** *Let* $\phi^*(s) = \max_{t\geq 0} ts - \phi(t)$. *Then we have*

$$\max_{\mathbf{p}\in\Delta_n} \sum_{i=1}^n p_i \ell(\mathbf{w}; \mathbf{z}_i) - \tau D_\phi\left(\mathbf{p} \,\|\, \frac{\mathbf{1}}{n}\right) = \min_\nu \frac{\tau}{n} \sum_{i=1}^n \phi^*\left(\frac{\ell(\mathbf{w}; \mathbf{z}_i) - \nu}{\tau}\right) + \nu. \tag{2.13}$$

The proof can be found in Example 1.14 in Chapter 1.

Examples of Regularized DRO

Example 2.1. (KL-divergence Regularized DRO) *For the special case of using KL-divergence, we can further simplify the above objective function. Since* $\phi(t) = t\log t - t + 1$, *then* $\phi^*(s) = \exp(s) - 1$ *(see Example 1.15) and solving* ν *yields*

$$\nu = \tau \log\left(\frac{1}{n}\sum_{i=1}^n \exp(\ell(\mathbf{w}; \mathbf{z}_i)/\tau)\right).$$

Plugging it back into the objective, we can obtain a simplified form

$$\max_{\mathbf{p}\in\Delta_n} \sum_{i=1}^n p_i \ell(\mathbf{w}; \mathbf{z}_i) - \tau KL\left(\mathbf{p}, \frac{\mathbf{1}}{n}\right) = \tau \log\left(\frac{1}{n}\sum_{i=1}^n \exp(\ell(\mathbf{w}; \mathbf{z}_i)/\tau)\right).$$

As a result, with $\phi(t) = t\log t - t + 1$, *the KL-divergence regularized DRO (2.11) is equivalent to*

$$\min_{\mathbf{w}} \tau \log\left(\frac{1}{n}\sum_{i=1}^n \exp\left(\frac{\ell(\mathbf{w}; \mathbf{z}_i)}{\tau}\right)\right). \tag{2.14}$$

Example 2.2. (Empirical CVaR) *As another example, we derive the dual form of the empirical CVaR. With simple algebra, we can derive that* $\phi^*(s) = \frac{[s]_+}{\alpha}$ *(see Example 1.15) for* $\phi(t) = \mathbb{I}_{0-\infty}(t \leq 1/\alpha)$.

As a result, with $\phi(t) = \mathbb{I}_{0-\infty}(t \le 1/\alpha)$, the regularized DRO (2.11) corresponding to the empirical CVaR$-\alpha$ is equivalent to

$$\min_{\mathbf{w},\nu} \frac{1}{n\alpha} \sum_{i=1}^{n} [\ell(\mathbf{w}; \mathbf{z}_i) - \nu]_+ + \nu. \tag{2.15}$$

When $k = n\alpha \in [1, n]$ is an integer, the above objective reduces to the average of top-k loss values when sorting them in descending order, as shown in the following lemma.

Lemma 2.2 *Let $\ell_{[i]}$ denote the i-th largest loss among $\{\ell(\mathbf{w}; \mathbf{z}_i), i = 1, \ldots, n\}$ ranked in descending order. If $\alpha = k/n$, we have*

$$\min_{\nu} \frac{1}{n\alpha} \sum_{i=1}^{n} [\ell(\mathbf{w}; \mathbf{z}_i) - \nu]_+ + \nu = \frac{1}{k} \sum_{i=1}^{k} \ell_{[i]}. \tag{2.16}$$

Proof. First, we have

$$\min_{\nu} \frac{1}{n\alpha} \sum_{i=1}^{n} [\ell(\mathbf{w}; \mathbf{z}_i) - \nu]_+ + \nu = \min_{\nu} \frac{1}{n\alpha} \sum_{i=1}^{n} [\ell_{[i]} - \nu]_+ + \nu.$$

Let ν_* be an optimal solution given $\mathbf{w}$. Due to the first-order optimality condition, we have

$$0 \in \frac{1}{k} \sum_{i=1}^{n} \partial_\nu [\ell_{[i]} - \nu_*]_+ + 1.$$

Hence,

$$-k \in \sum_{i=1}^{n} \partial_\nu [\ell_{[i]} - \nu_*]_+. \tag{2.17}$$

Let us first assume $\ell_{[k+1]} < \ell_{[k]}$. We will show hat $\nu_* \in (\ell_{[k+1]}, \ell_{[k]}]$ satisfy this condition. Since $-1 \in \partial_\nu [\ell_{[i]} - \nu_*]_+$ for $i = 1, \ldots, k$ due to $\ell_{[i]} \ge \nu_*$ and $\partial_\nu [\ell_{[i]} - \nu_*]_+ = 0$ for $i = k+1, \ldots, n$ due to $\ell_{[i]} < \nu_*$. Hence, it verifies that the condition (2.17) holds at such ν_*.

If $\ell_{[k+1]} = \ell_{[k]}$, we argue that $\nu_* = \ell_{[k]}$ can still satisfy (2.17). This is because $-1 \in \partial_\nu [\ell_{[i]} - \nu_*]_+$ for $i = 1, \ldots, k$ and $0 \in \partial_\nu [\ell_{[i]} - \nu_*]_+$ for $\ell_{[i]} = \ell_{[k+1]}, i \ge k+1$ and $\partial_\nu [\ell_{[i]} - \nu_*]_+ = 0$ for $\ell_{[i]} < \ell_{[k+1]}, i \ge k+1$. Then the conclusion follows. □

The Dual form of Constrained DRO

For transforming the constrained DRO, we can use the following proposition based on the Lagrangian duality theory.

Proposition 2.2 (Dual form of Constrained DRO)**.** *Let* $\phi^*(s) = \max_{t\geq 0} ts - \phi(t)$. *Then we have*

$$\max_{\mathbf{p}\in\Delta_n, D_\phi(\mathbf{p}\,\|\,\frac{1}{n})\leq\rho} \sum_{i=1}^n p_i \ell(\mathbf{w};\mathbf{z}_i) = \min_{\tau\geq 0,\nu} \frac{\tau}{n}\sum_{i=1}^n \phi^*\left(\frac{\ell(\mathbf{w};\mathbf{z}_i)-\nu}{\tau}\right) + \nu + \tau\rho. \quad (2.18)$$

The proof is similar to that of Proposition 2.1.

Examples of Constrained DRO

Example 2.3. *(***KL Constrained DRO***)* *With* $\phi(t) = t\log t - t + 1$, *the KL-divergence constrained DRO (2.12) is equivalent to:*

$$\min_{\mathbf{w},\tau\geq 0} \tau\log\left(\frac{1}{n}\sum_{i=1}^n \exp\left(\frac{\ell(\mathbf{w};\mathbf{z}_i)}{\tau}\right)\right) + \tau\rho. \quad (2.19)$$

KL-regularized DRO and KL-constrained DRO play important roles in many modern artificial intelligence applications. The LDR loss (2.5) can be interpreted as a form of KL-regularized DRO, except that the uncertainty is placed on the distribution of class labels for each individual data point. We will present additional applications in Section 2.4.

The Optimization Challenge

Although the transformed optimization problems do not involve dealing with a high-dimensional variable $\mathbf{p} \in \Delta_n$, the new optimization problems (2.14), (2.19) are not of the same form as ERM. The critical assumption that an unbiased gradient can be easily computed fails. We will cast them as instances of stochastic compositional optimization (SCO), which is topic of Chapter 4 of the book.

2.2.2 Optimized Certainty Equivalent

How to understand the generalization of DRO? One way is to still consider bounding the expected risk $\mathcal{R}(\mathbf{w})$ of the learned model. However, the expected risk may not be a good measure when the data distribution is skewed.

For simplicity, let us consider a binary classification problem with $\Pr(\mathbf{x}, y=1) = \pi_+ \Pr(\mathbf{x}|y=1)$ and $\Pr(\mathbf{x}, y=-1) = \pi_- \Pr(\mathbf{x}|y=-1)$, where $\pi_+ = \Pr(y=1), \pi_- = \Pr(y=-1)$. Let $\mathbb{P}_+$ and $\mathbb{P}_-$ be the distributions of $\mathbf{x}$ conditioned on $y=1$ and $y=-1$, respectively. By the law of total expectation we have

$$\mathcal{R}(\mathbf{w}) = \mathbb{E}_{\mathbf{x},y}\ell(h(\mathbf{w};\mathbf{x}),y) = \pi_+\mathbb{E}_{\mathbf{x}\sim\mathbb{P}_+}[\ell(h(\mathbf{w};\mathbf{x}),1)] + \pi_-\mathbb{E}_{\mathbf{x}\sim\mathbb{P}_-}[\ell(h(\mathbf{w};\mathbf{x}),-1)]. \quad (2.20)$$

If $\pi_- \gg \pi_+$, the expected risk would be dominated by the expected loss of data from the negative class. As a result, a small $\mathcal{R}(\mathbf{w})$ does not necessarily indicate a small $\mathbb{E}_{\mathbf{x}\sim\mathbb{P}_+}[\ell(\mathbf{w};\mathbf{x},1)]$.

Instead, we consider the population risk of DRO as the target measure. A formal definition of the population risk for the regularized DRO (2.11) is given below.

Definition 2.4 (Population risk of DRO) Given a data distribution $\mathbb{P}$, for any $\tau > 0$, we define the population risk of regularized DRO (2.11) as:

$$\mathcal{R}_{\text{oce}}(\mathbf{w}) := \max_{\mathbb{Q}\in\mathcal{Q}} \mathbb{E}_{\mathbf{z}'\sim\mathbb{Q}}\ell(\mathbf{w};\mathbf{z}') - \tau\mathbb{E}_{\mathbb{P}}\phi\left(\frac{d\mathbb{Q}}{d\mathbb{P}}\right) \tag{2.21}$$

$$= \min_{\nu} \tau\mathbb{E}_{\mathbf{z}\sim\mathbb{P}}\phi^*\left(\frac{\ell(\mathbf{w};\mathbf{z})-\nu}{\tau}\right) + \nu, \tag{2.22}$$

where $\phi^*(s) = \max_{t\geq 0} ts - \phi(t)$.

In the definition above, $\mathcal{Q} = \{\mathbb{Q} \mid \mathbb{Q} \ll \mathbb{P}\}$ denotes the set of probability measures that are absolutely continuous with respect to $\mathbb{P}$. A probability measure $\mathbb{Q}$ is said to be absolutely continuous with respect to $\mathbb{P}$, denoted $\mathbb{Q} \ll \mathbb{P}$, if every event that has probability 0 under $\mathbb{P}$ also has probability 0 under $\mathbb{Q}$. If $\mathbb{P}$ and $\mathbb{Q}$ admit densities $p(z)$ and $q(z)$ with respect to a common dominating measure on $\mathcal{Z}$, and $\mathbb{Q} \ll \mathbb{P}$, then

$$\mathbb{E}_{\mathbb{P}}\left[\phi\left(\frac{d\mathbb{Q}}{d\mathbb{P}}\right)\right] = \int_{\mathcal{Z}} p(z)\,\phi\left(\frac{q(z)}{p(z)}\right)\,dz.$$

The equivalent counterpart in (2.22) is a risk measure originates from the **optimized certainty equivalent (OCE)**, a concept popularized in mathematical economics (Ben-Tal and Teboulle, 1986a). Minimizing OCE has an effect of so-called **risk-aversion**, which discourages models from having rare but catastrophic errors. Two special cases are discussed below:

- When $\phi(t) = \mathbb{I}_{0-\infty}(t \leq 1/\alpha)$, the OCE becomes the CVaR-α, i.e.,

  $$\mathcal{R}_{\text{cvar}}(\mathbf{w}) = \mathbb{E}_{\mathbf{z}}[\ell(\mathbf{w};\mathbf{z})|\ell(\mathbf{w};\mathbf{z}) \geq \text{VAR}_\alpha(\ell(\mathbf{w};\mathbf{z}))],$$

  where $\text{VAR}_\alpha(\ell(\mathbf{w};\mathbf{z})) = \sup_s[\Pr(\ell(\mathbf{w};\mathbf{z}) \geq s) \geq \alpha]$ is the α-quantile or "value-at-risk" of the random loss values.
- When $\phi(t) = t\log t - t + 1$, OCE becomes the entropic risk:

  $$\mathcal{R}_{\text{ent}}(\mathbf{w}) = \tau\log\left(\mathbb{E}_{\mathbf{z}}\exp\left(\frac{\ell(\mathbf{w};\mathbf{z})}{\tau}\right)\right).$$

What is risk-aversion?

Risk aversion refers to the preference for a certain and predictable cost over an uncertain outcome with the same average cost, especially when the uncertainty involves rare but severe losses. This behavior cannot be captured by the expectation alone, which treats all outcomes linearly and ignores tail risk.

The OCE provides a principled risk-sensitive alternative by assigning a single certainty-equivalent value to a random loss that accounts for both its mean and its variability. A classic illustration is insurance: consider paying a fixed premium of \$1,000 versus facing a \$100,000 medical bill with probability 0.01 and zero cost otherwise. Although both options have the same expected cost, the OCE risk ($\log \mathbb{E}[\exp(X)]$ assigns a much larger value to the uninsured option, as it heavily penalizes the rare catastrophic loss. Consequently, OCE correctly reflects the economic rationale behind insurance decisions by favoring stable outcomes over risky alternatives with heavy tails.

We present two properties of OCE below.

Lemma 2.3 *Let* $\partial\phi^*(t) = \{s : \phi'^*_-(t) \le s \le \phi'^*_+(t)\}$. *If* $a < b$, *then* $0 \le \phi'^*_+(a) \le \phi'^*_-(b)$.

Proof. Due to the definition $\phi^*(s) = \max_{t\ge 0} ts - \phi(t)$, we have $\partial\phi^*(s) \ge 0$, which indicates that ϕ^* is non-decreasing. Since ϕ^* is also convex, the conclusion follows from the convex analysis (Rockafellar, 1970b)[Section 24]. □

Lemma 2.4 *For any* $\tau > 0, \mathbf{w} \in \mathbb{R}^d$, *it holds that* $\mathcal{R}_{oce}(\mathbf{w}) \ge \mathcal{R}(\mathbf{w})$.

Proof. Since $\phi(1) = 0$, then $\phi^*(s) = \max_{t\ge 0} ts - \phi(t) \ge s - \phi(1) = s$. Hence,

$$\begin{aligned}\mathcal{R}_{\text{oce}}(\mathbf{w}) &= \min_{\nu} \tau \mathbb{E}_{\mathbf{z}} \phi^*\left(\frac{\ell(\mathbf{w};\mathbf{z}) - \nu}{\tau}\right) + \nu \\ &\ge \min_{\nu} \tau \mathbb{E}_{\mathbf{z}}\left(\frac{\ell(\mathbf{w};\mathbf{z}) - \nu}{\tau}\right) + \nu = \mathcal{R}(\mathbf{w}).\end{aligned}$$

□

Why it matters:

Lemma 2.3 implies that a data with a larger loss $\ell(h(\mathbf{w};\mathbf{x}), y)$ will have a higher weight in the gradient calculation in terms of $\mathbf{w}$.
Lemma 2.4 indicates that OCE is a stronger measure than the expected risk. A small OCE will imply a small expected risk, while the reverse is not necessarily true.

Based on OCE, we can define the excess risk $\mathcal{R}_{\text{oce}}(\mathbf{w}) - \min_{\mathbf{u}\in\mathcal{W}} \mathcal{R}_{\text{oce}}(\mathbf{u})$ and decompose it into an optimization error and a generalization error similar to Lemma 2.1.

Lemma 2.5 *For a learned model* $\mathbf{w} = \mathcal{A}(\mathcal{S};\zeta)$ *for solving empirical DRO (2.11), we have*

$$\mathcal{R}_{oce}(\mathbf{w}) - \min_{\mathbf{u}\in\mathcal{W}} \mathcal{R}_{oce}(\mathbf{u}) \le 2 \underbrace{\sup_{\mathbf{w}\in\mathcal{W}} |\mathcal{R}_{oce}(\mathbf{w}) - \hat{\mathcal{R}}_{\mathcal{S}}(\mathbf{w})|}_{\textit{generalization error}} + \underbrace{\hat{\mathcal{R}}_{\mathcal{S}}(\mathbf{w}) - \min_{\mathbf{u}\in\mathcal{W}} \hat{\mathcal{R}}_{\mathcal{S}}(\mathbf{w})}_{\textit{optimization error}}.$$

Fig. 2.2: Illustrative of spurious correlation between the class label and some feature: waterbird images mostly have water background and landbird images mostly have land background.

2.2.3 Group Distributionally Robust Optimization

Group DRO is an extension of DRO by aggregating data into groups and using DRO on the group level to formulate a robust risk function. It is helpful to promote equity of the learned model and mitigating the impact of spurious correlations that exist between the label and some features, by using prior knowledge to group the data.

Let us consider an illustrative example of classifying waterbird images from landbird images (see Figure 2.2). The training data may have the same number of waterbird images and landbird images. However, most waterbird images may have water in the background and most landbird images may have land in the background. Standard empirical risk minimization may learn spurious correlation between the class labels (e.g., waterbird) and the specific value of some attribute (e.g., the water background). As a consequence, the model may perform poorly on waterbird images with land background.

Critical: Data may exhibit imbalance not in the marginal distribution of class label but some joint distribution of the class label and some attributes, which causes the spurious correlation.

GDRO can be used to mitigate this issue by leveraging prior knowledge of spurious correlations to define groups over the training data. Let the training data be divided into multiple groups $\mathcal{G}_1, \ldots, \mathcal{G}_K$, where $\mathcal{G}_j = \{(\mathbf{x}_1^j, y_1^j), \ldots (\mathbf{x}_{n_j}^j, y_{n_j}^j)\}$ includes a set of examples from the j-th group. We define an averaged loss over examples from each group $L_j(\mathbf{w}) = \frac{1}{n_j} \sum_{i=1}^{n_j} \ell(h(\mathbf{w}; \mathbf{x}_i^j), y_i^j)$. Then, a regularized group DRO can be defined as

$$\min_{\mathbf{w}} \max_{\mathbf{p} \in \Delta_K} \sum_{j=1}^{K} p_j L_j(\mathbf{w}) - \tau D_\phi \left(\mathbf{p} \,\|\, \frac{\mathbf{1}}{K} \right), \tag{2.23}$$

and a constrained group DRO is given by:

$$\min_{\mathbf{w}} \max_{\mathbf{p} \in \Delta_K, D_\phi(\mathbf{p} \| \frac{\mathbf{1}}{K}) \leq \rho} \sum_{j=1}^{K} p_j L_j(\mathbf{w}). \tag{2.24}$$

By doing so, the learning process is less likely to be dominated by the majority group associated with the spurious correlation between the label and a particular feature (e.g., waterbird images with water background). If the model only captures the spurious correlation, the loss for the minority group will be large, which in turn drives the learning process to reduce this loss and thereby mitigate the spurious correlation.

Examples and Reformulations

Similar to before, we can convert the min-max problem into a minimization problem to reduce additional overhead of dealing with a large number of groups. We give two examples of using the KL-divergence constraint of $\mathbf{p}$ and CVaR-α.

With $\phi(t) = t \log t - t + 1$, the KL-divergence constrained group DRO (2.24) is equivalent to

$$\min_{\mathbf{w}, \tau \geq 0} \tau \log \left(\frac{1}{K} \sum_{j=1}^{K} \exp\left(\frac{L_j(\mathbf{w})}{\tau} \right) \right) + \tau \rho. \tag{2.25}$$

With $\phi(t) = \mathbb{I}_{0-\infty}(t \leq 1/\alpha)$, CVaR-$\alpha$ group DRO (2.24) is equivalent to

$$\min_{\mathbf{w}, \nu} \frac{1}{K\alpha} \sum_{j=1}^{K} [L_j(\mathbf{w}) - \nu]_+ + \nu. \tag{2.26}$$

The Optimization Challenge

Again, these new optimization problems (2.25), (2.26) cannot be solved by simply using existing stochastic algorithms for ERM since $L_j(\mathbf{w})$ depends on many data and they are inside non-linear functions. In particular, the problem (2.26) is an instance of finite-sum coupled compositional optimization (FCCO), which will be explored in Chapter 5 in depth.

2.3 Empirical X-risk Minimization

So far, we have revisited classical ideas of machine learning based on empirical risk minimization and its distributionally robust variants. In these risk functions, we assume each data defines a loss based on itself. These losses are typically surrogate functions of a prediction error measuring the inconsistency between the prediction and the label.

However, such loss functions are insufficient to capture many objectives, which involve comparison between different data points. Examples include areas under ROC curves (AUROC) and areas under precision-recall curves (AUPRC) for imbal-

anced data classification, ranking measures such as normalized discounted cumulative gain (NDCG), mean average precision (MAP) and listwise losses for learning to rank, and contrastive losses for representation learning.

The standard ERM framework is inadequate for optimizing such metrics and losses, as they involve interactions across multiple data points. We need a new mathematical framework to understand the challenge and to design provable and practical algorithms. To this end, we introduce a new risk minimization framework, named empirical X-risk minimization (EXM), as defined below:

Empirical X-risk Minimization (EXM)

X-risk refers to a family of risks such that the loss of each data is defined in a way that contrasts the data with many others. Mathematically, empirical X-risk minimization is formulated as:

$$\min_{\mathbf{w}} \frac{1}{n} \sum_{i=1}^{n} f_i(g(\mathbf{w}, \mathbf{x}_i, \mathcal{S}_i)), \tag{2.27}$$

where $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ is a set of data points, each $\mathcal{S}_i$ contains a number of items, f_i is a simple but non-linear function, and $g(\mathbf{w}, \mathbf{x}_i, \mathcal{S}_i)$ involves the coupling between $\mathbf{x}_i$ and all data in $\mathcal{S}_i$. A simple instance of $g(\mathbf{w}, \mathbf{x}_i, \mathcal{S}_i)$ is the following averaged form:

$$g(\mathbf{w}, \mathbf{x}_i, \mathcal{S}_i) = \frac{1}{|\mathcal{S}_i|} \sum_{\mathbf{z} \in \mathcal{S}_i} \ell(\mathbf{w}; \mathbf{x}_i, \mathbf{z}). \tag{2.28}$$

With g given in (2.28), EXM is an instance of finite-sum coupled compositional optimization (**FCCO**), a framework explored in detail in Chapter 5.

Below, we present several important instances of X-risks.

2.3.1 AUC Losses

AUC, short for Area under receiver operating characteristic (ROC) curve, is commonly used to measure performance for the imbalanced data classification.

What is Imbalanced Data Classification?

Imbalanced data classification refers to classification problems, where the number of examples from some classes is significantly larger than that of other classes.

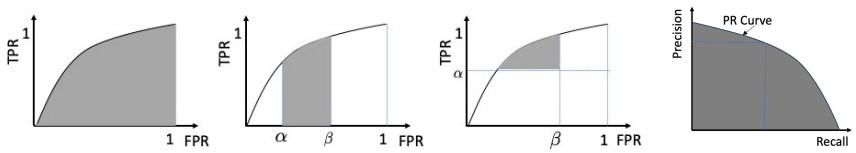


Fig. 2.3: Areas under ROC Curves

Definition and an Empirical Estimator of AUC

The ROC curve is the plot of the true positive rate (TPR) against the false positive rate (FPR) at each threshold setting. Let $\mathbb{P}_+, \mathbb{P}_-$ denote the distribution of random positive and negative data, respectively. Let $h(\cdot) : \mathcal{X} \to \mathbb{R}$ denote a predictive scoring function. For a given threshold t, the TPR of h can be written as $\text{TPR}(t) = \Pr(h(\mathbf{x}) > t|y = 1) = \mathbb{E}_{\mathbf{x}\sim\mathbb{P}_+}[\mathbb{I}(h(\mathbf{x}) > t)]$, and the FPR can be written as $\text{FPR}(t) = \Pr(h(\mathbf{x}) > t|y = -1) = \mathbb{E}_{\mathbf{x}\sim\mathbb{P}_-}[\mathbb{I}(h(\mathbf{x}) > t)]$. Let $F_-(t) = 1 - \text{FPR}(t)$ denote the cumulative density function of the random variable $h(\mathbf{x}_-)$ for $\mathbf{x}_- \sim \mathbb{P}_-$. Let $p_-(t)$ denote its corresponding probability density function. Similarly, let $F_+(t) = 1 - \text{TPR}(t)$ and $p_+(t)$ denote the cumulative density function and the probability density function of $h(\mathbf{x}_+)$ for $\mathbf{x}_+ \sim \mathbb{P}_+$, respectively.

For a given $u \in [0, 1]$, let $\text{FPR}^{-1}(u) = \inf\{t \in \mathbb{R} : \text{FPR}(t) \leq u\}$. The ROC curve is defined as $\{u, \text{ROC}(u)\}$, where $u \in [0, 1]$ and $\text{ROC}(u) = \text{TPR}(\text{FPR}^{-1}(u))$.

Hence, we have the following theorem.

Theorem 2.1 *The AUC for a predictive scoring function h is equal to*

$$AUC(h) = \Pr(h(\mathbf{x}_+) > h(\mathbf{x}_-)) = \mathbb{E}_{\mathbf{x}_+\sim\mathbb{P}_+,\mathbf{x}_-\sim\mathbb{P}_-}[\mathbb{I}(h(\mathbf{x}_+) > h(\mathbf{x}_-))]. \tag{2.29}$$

Proof. The AUC score of h is given by

$$\begin{aligned}\text{AUC}(h) &= \int_0^1 \text{ROC}(u)du = \int_{-\infty}^{\infty} \text{TPR}(t)dF_-(t) = \int_{-\infty}^{\infty} \text{TPR}(t)p_-(t)dt \\ &= \int_{-\infty}^{\infty}\int_t^{\infty} p_+(s)dsp_-(t)dt = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} p_+(s)p_-(t)\mathbb{I}(s > t)dsdt.\end{aligned}$$

Since $h(\mathbf{x}_+)$ follows $p_+(s)$ and $h(\mathbf{x}_-)$ follows $p_-(t)$, we can conclude the proof. □

This indicates that AUC is a pairwise ranking metric. An ideal scoring function that ranks all positive examples above negative examples has a perfect AUC score 1. It also implies the following empirical non-parametric estimator of AUC based on a set of data $\mathcal{S}$ with n_+ positive samples in $\mathcal{S}_+$ and n_- negative samples in $\mathcal{S}_-$:

$$\text{AUC}(h; \mathcal{S}) = \frac{1}{n_+n_-} \sum_{\mathbf{x}_+\in\mathcal{S}_+,\mathbf{x}_-\in\mathcal{S}_-} \mathbb{I}(h(\mathbf{x}_+) > h(\mathbf{x}_-)), \tag{2.30}$$

which is also known as the Mann-Whitney U-statistic (Hanley and McNeil, 1982).

Necessity of Maximizing AUC

AUC is more appropriate than accuracy for assessing the performance of imbalanced data classification. Let us consider an example with 2 positive data and 100 negative data. If one positive data has a prediction score 0.5 and another one has a prediction score -0.2, and all negative data has prediction scores less than 0 but larger than -0.2. In this case, if we choose a classification threshold as 0, then the accuracy is $101/102 = 0.99$. However, the (emprical) AUC score according to (2.30) is given by $100/200 = 0.5$. "*Can a model that optimizes the accuracy also optimize the AUC score?*" Unfortunately, this is not the case as different classifiers that have the same accuracy could have dramatic different AUC (Cortes and Mohri, 2003). An example is illustrated in Table 2.2. Hence, it makes sense to directly optimize AUC.

Critical: A model that optimizes accuracy does not necessarily optimize AUC.

Example 1		2c**Example 2**		2c**Example 3**	
Prediction	Ground Truth	Prediction	Ground Truth	Prediction	Ground Truth
0.9	1	0.9	1	0.9	1
0.8	1	**0.41**(↓)	1	**0.41**(↓)	1
0.7	1	0.7	1	**0.40**(↓)	1
0.6	0	0.6	0	**0.49**(↓)	0
0.6	0	**0.49**(↓)	0	**0.48**(↓)	0
0.47	0	0.47	0	0.47	0
0.47	0	0.47	0	0.47	0
0.45	0	0.45	0	0.45	0
0.43	0	0.43	0	0.43	0
0.42	0	0.42	0	0.42	0
⋮	⋮	⋮	⋮	⋮	⋮
0.1	0	0.1	0	0.1	0
Acc=0.92		Acc=0.92 (—)		Acc=0.92 (—)	
AUC=1.00		AUC=**0.89** (↓)		AUC=**0.78** (↓)	

Table 2.2: Illustrations of variance of AUC for different classifiers with the same Accuracy on an imbalanced dataset of 25 samples with a positive ratio of 3/25. The accuracy threshold is 0.5. **Example 1** shows that all positive instances rank higher than negative instances and two negative instances are misclassified to positive class. **Example 2** shows that 1 positive instance ranks lower than 7 negative instances and 1 positive and 1 negative instances are missclassifed. **Example 3** shows that 2 positive instances rank lower than 7 negative instances, and 2 positive instances are also missclassifed as negative class. Overall, we can observe that AUC drops dramatically as the ranks of positive instances drop but meanwhile Accuracy remains unchanged.

Pairwise Loss	$\ell(t)$	Monotone
Squared Hinge	$(c+t)_+^2$	Yes
Hinge	$(c+t)_+$	Yes
Logistic	$\log(1+\exp(st))$	Yes
Sigmoid	$(1+\exp(-st))^{-1}$	Yes
Square	$(c+t)^2$	No
Barrier Hinge	$\max(-s(c-t)+c, \max(s(-t-c), c+t))$	No

Table 2.3: Surrogate loss functions for pairwise modeling with the input argument $t = h(\mathbf{w}; \mathbf{x}_-) - h(\mathbf{w}; \mathbf{x}_+)$. For the sake of simplicity, denote $\max(0, t)$ by t_+, denote the scaling hyper-parameter by $s > 0$ and margin hyper-parameter by $c > 0$.

Pairwise Surrogate Losses

Using a pairwise surrogate loss $\ell(\cdot)$ of the indicator function $\mathbb{I}(t \geq 0)$ (see examples in Table 2.3), we have the following empirical AUC optimization problem for learning a parameterized function $h(\mathbf{w}; \cdot)$:

$$\min_{\mathbf{w}\in\mathbb{R}^d} \frac{1}{n_+}\frac{1}{n_-} \sum_{\mathbf{x}_i\in\mathcal{S}_+} \sum_{\mathbf{x}_j\in\mathcal{S}_-} \ell(h(\mathbf{w}; \mathbf{x}_j) - h(\mathbf{w}; \mathbf{x}_i)). \tag{2.31}$$

This can be regarded as a special case of (2.27) by setting

$$g(\mathbf{w}; \mathbf{x}_i, \mathcal{S}_-) = \frac{1}{n_-} \sum_{\mathbf{x}_j\in\mathcal{S}_-} \ell(h(\mathbf{w}; \mathbf{x}_j) - h(\mathbf{w}; \mathbf{x}_i)),$$
$$f_i(g) = g.$$

This is the simplest form of EXM as f is just a linear function. An unbiased stochastic gradient can be easily computed based on a pair of data points consisting of a random positive and a random negative data point.

Compositional Objectives

An alternative approach to formulate AUC maximization is to decouple the pairwise comparison between positive and negative examples. A generic formulation is given by:

$$\begin{aligned}\min_{\mathbf{w}\in\mathbb{R}^d,(a,b)\in\mathbb{R}^2} \quad & \frac{1}{|\mathcal{S}_+|} \sum_{\mathbf{x}_i\in\mathcal{S}_+} (h(\mathbf{w}; \mathbf{x}_i) - a)^2 + \frac{1}{|\mathcal{S}_-|} \sum_{\mathbf{x}_j\in\mathcal{S}_-} (h(\mathbf{w}; \mathbf{x}_j) - b)^2 \\ & + f\left(\frac{1}{|\mathcal{S}_-|} \sum_{\mathbf{x}_j\in\mathcal{S}_-} h(\mathbf{w}; \mathbf{x}_j) - \frac{1}{|\mathcal{S}_+|} \sum_{\mathbf{x}_i\in\mathcal{S}_+} h(\mathbf{w}; \mathbf{x}_i)\right),\end{aligned} \tag{2.32}$$

where f is a non-linear function. The last component is a compositional function.

The above formulation also has a clear physical meaning. In particular, minimizing the first two terms aim to push the prediction scores of positive and negative examples to center around their means, respectively, and minimizing the third term aims to push the mean score of positive examples to be larger than the mean score of negative examples.

The above formulation is motivated by the pairwise formulation with a square surrogate function $\ell(h(\mathbf{w};\mathbf{x}_j) - h(\mathbf{w};\mathbf{x}_i)) = (c + h(\mathbf{w};\mathbf{x}_j) - h(\mathbf{w};\mathbf{x}_i))^2$. Indeed, in this case, (2.31) is equivalent to (2.32) with $f(s) = (s+c)^2$. We leave this as an exercise for interested readers. Nevertheless, using $f(s) = [s+c]_+^2$ in (2.32) is more robust than $f(s) = (s+c)^2$ with $c > 0$.

Solving the above problem requires compositional optimization techniques, which will be discussed in Section 6.4.1.

2.3.2 Average Precision Loss

Area under precision-recall curve (AUPRC) is another commonly used measure for highly imbalanced data. The precision and recall of a scoring function h at threshold t are defined as

$$\begin{aligned}\mathrm{Rec}(t) &:= \Pr(h(\mathbf{x}) > t \mid y = 1) = \mathrm{TPR}(t),\\ \mathrm{Prec}(t) &:= \Pr(y = 1 \mid h(\mathbf{x}) > t).\end{aligned}$$

For a given $u \in [0,1]$, let $\mathrm{TPR}^{-1}(u) = \inf\{t \in \mathbb{R} : \mathrm{TPR}(t) \le u\}$. The precision–recall (PR) curve is defined as $\{(u, \mathrm{PR}(u))\}$, where $u \in [0,1]$ and $\mathrm{PR}(u) = \mathrm{Prec}\big(\mathrm{TPR}^{-1}(u)\big)$. Hence, AUPRC for h can be computed by

$$\mathrm{AUPRC}(h) = \int_0^1 \mathrm{PR}(u)\, du.$$

Theorem 2.2 *The AUPRC for a predictive scoring function h is equal to*

$$AUPRC(h) = \int_{-\infty}^{\infty} Prec(t)\, p_+(t)\, dt = \mathbb{E}_{\mathbf{x}_+ \sim \mathbb{P}_+}\big[Prec(h(\mathbf{x}_+))\big]. \tag{2.33}$$

Proof. By definition,

$$\mathrm{AUPRC}(h) = \int_0^1 \mathrm{PR}(u)\, du = \int_0^1 \mathrm{Prec}\big(\mathrm{TPR}^{-1}(u)\big)\, du.$$

Let $u = \mathrm{TPR}(t) = 1 - F_+(t)$. Then $du = -p_+(t)\, dt$. Therefore,

$$\mathrm{AUPRC}(h) = \int_{\infty}^{-\infty} \mathrm{Prec}(t)\, (-p_+(t)\, dt) = \int_{-\infty}^{\infty} \mathrm{Prec}(t)\, p_+(t)\, dt,$$

which proves (2.33). □

The above theorem yields the following empirical estimator of AUPRC. For a set of training examples $\mathcal{S} = \mathcal{S}_+ \cup \mathcal{S}_-$, a non-parametric estimator of AUPRC is average precision (AP) (Boyd et al., 2013):

$$\mathrm{AP}(h) = \frac{1}{n_+} \sum_{\mathbf{x}_i \in \mathcal{S}_+} \frac{\sum\limits_{\mathbf{x}_j \in \mathcal{S}_+} \mathbb{I}(h(\mathbf{x}_j) \geq h(\mathbf{x}_i))}{\sum\limits_{\mathbf{x}_j \in \mathcal{S}} \mathbb{I}(h(\mathbf{x}_j) \geq h(\mathbf{x}_i))}. \tag{2.34}$$

AP is an unbiased estimator of AUPRC in the limit $n \to \infty$.

Necessity of Maximizing AUPRC

While AUC is generally more suitable than accuracy for imbalanced classification tasks, it may fail to adequately capture misorderings among top-ranked examples. Consider a scenario with 2 positive and 100 negative samples. If the two positive samples are ranked below just two of the negative ones, followed by the remaining 98 negatives, the resulting AUC is $196/200 = 0.98$, which appears high. However, this model would be inadequate if our focus is on the top two predicted positive instances. In drug discovery, for example, models are expected to identify the most promising candidate molecules for experimental validation. If these top-ranked predictions turn out to lack the desired properties, the resulting experimental efforts may lead to significant wasted resources and costly failures.

To avoid this issue, AUPRC or its empirical estimator AP is typically used as a performance metric. According to its definition (2.34), the AP score for the above example is $\frac{1}{2}(\frac{1}{3} + \frac{2}{4}) = 0.42$. In contrast, a perfect ranking that ranks the two positive examples at the top gives an AP score of 1. Unfortunately, optimizing AUC does not necessarily lead to optimal AP, as two models with identical AUC scores can exhibit significantly different AP values. This highlights the need for efficient optimization algorithms that directly maximize AP.

Critical: AUPRC/AP penalizes more on the error at the top of the ranked list.

Surrogate Loss of AP

To construct a differentiable objective for minimization, a differentiable surrogate loss $\ell(h(\mathbf{x}_j) - h(\mathbf{x}_i))$ is used in place of $\mathbb{I}(h(\mathbf{x}_j) \geq h(\mathbf{x}_i))$. Then AP can be approximated by :

$$\text{AP} \approx \frac{1}{n_+} \sum_{\mathbf{x}_i \in \mathcal{S}_+} \frac{\sum_{\mathbf{x}_j \in \mathcal{S}} \mathbb{I}(y_j = 1)\ell(h(\mathbf{x}_j) - h(\mathbf{x}_i))}{\sum_{\mathbf{x}_j \in \mathcal{S}} \ell(h(\mathbf{x}_j) - h(\mathbf{x}_i))}. \tag{2.35}$$

Let us define

$$\begin{aligned}
&f(\mathbf{g}) = -\frac{[\mathbf{g}]_1}{[\mathbf{g}]_2}, \\
&\mathbf{g}(\mathbf{w}; \mathbf{x}_i, \mathcal{S}) = [g_1(\mathbf{w}; \mathbf{x}_i, \mathcal{S}), g_2(\mathbf{w}; \mathbf{x}_i, \mathcal{S})] \\
&g_1(\mathbf{w}; \mathbf{x}_i, \mathcal{S}) = \frac{1}{|\mathcal{S}|} \sum_{\mathbf{x}_j \in \mathcal{S}} \mathbb{I}(y_j = 1)\ell(h(\mathbf{w}; \mathbf{x}_j) - h(\mathbf{w}; \mathbf{x}_i)), \\
&g_2(\mathbf{w}; \mathbf{x}_i, \mathcal{S}) = \frac{1}{|\mathcal{S}|} \sum_{\mathbf{x}_j \in \mathcal{S}} \ell(h(\mathbf{w}; \mathbf{x}_j) - h(\mathbf{w}; \mathbf{x}_i)).
\end{aligned}$$

Then, we formulate AP maximization as the following problem:

$$\min_{\mathbf{w}} \frac{1}{n_+} \sum_{\mathbf{x}_i \in \mathcal{S}_+} f(\mathbf{g}(\mathbf{w}; \mathbf{x}_i, \mathcal{S})), \tag{2.36}$$

which is a special case of EXM. We will explore efficient algorithms for optimizing AP in Section 6.4.2 using FCCO techniques.

2.3.3 Partial AUC Losses

There are two commonly used versions of partial AUC (pAUC), namely one-way pAUC (OPAUC) and two-way pAUC (TPAUC). OPAUC puts a restriction on the range of FPR, i.e., FPR $\in [\alpha, \beta]$ (the second figure from the left in Figure 2.3) and TPAUC puts a restriction on the lower bound of TPR and the upper bound of FPR, i.e., TPR $\geq \alpha$, FPR $\leq \beta$ (the second figure from the right in Figure 2.3).

By the definition, we have the following probabilistic interpretations.

Theorem 2.3 *OPAUC with FPR restricted in the range* $[\alpha, \beta]$ *for a predictive scoring function* h *is equal to*

$$OPAUC(h|FPR \in (\alpha, \beta)) = \Pr(h(\mathbf{x}_+) > h(\mathbf{x}_-), h(\mathbf{x}_-) \in [FPR^{-1}(\beta), FPR^{-1}(\alpha)]). \tag{2.37}$$

Similarly, TPAUC with FPR restricted in a range of $[0, \beta]$ *and TPR restricted in a range of* $[\alpha, 1]$ *is equal to*

$$\begin{aligned}
&TPAUC(h|TPR \geq \alpha, FPR \leq \beta) \\
&\quad = \Pr(h(\mathbf{x}_+) > h(\mathbf{x}_-), h(\mathbf{x}_-) \geq FPR^{-1}(\beta), h(\mathbf{x}_+) \leq TPR^{-1}(\alpha)\}).
\end{aligned} \tag{2.38}$$

Proof. The first part about OPAUC is similar to AUC except for the range of integral:

$$\begin{aligned}\mathrm{OPAUC}(h|\mathrm{FPR} \in (\alpha,\beta)) &= \int_{\mathrm{FPR}^{-1}(\beta)}^{\mathrm{FPR}^{-1}(\alpha)} \mathrm{TPR}(t)dF_-(t) \\ &= \int_{\mathrm{FPR}^{-1}(\beta)}^{\mathrm{FPR}^{-1}(\alpha)} \int_{-\infty}^{\infty} p_+(s)p_-(t)\mathbb{I}(s>t)dsdt.\end{aligned}$$

This concludes the proof of the first part.

For TPAUC with FPR restricted in $[0,\beta]$ and TPR restricted in $[\alpha,1]$, it is equal to OPAUC with FPR restricted in $[\gamma,\beta]$ minus the square area with FPR $\in [\gamma,\beta]$ and TPR $< \alpha$, where γ is the FPR that corresponds to TPR equals to α, i.e., $\mathrm{FPR}^{-1}(\gamma) = \mathrm{TPR}^{-1}(\alpha)$. Since $\mathrm{TPR}(t) = \int_t^\infty p_+(s)ds$ and $\mathrm{FPR}(t) = \int_t^\infty p_-(s)ds$, we have

$$\alpha = \int_{\mathrm{TPR}^{-1}(\alpha)}^{\infty} p_+(s)ds, \quad \beta = \int_{\mathrm{FPR}^{-1}(\beta)}^{\infty} p_-(t)dt.$$

Then, we have

$$\begin{aligned}&(\beta-\gamma)\alpha \\ &= \int_{\mathrm{FPR}^{-1}(\beta)}^{\infty}\int_{\mathrm{TPR}^{-1}(\alpha)}^{\infty} p_+(s)p_-(t)dsdt - \int_{\mathrm{FPR}^{-1}(\gamma)}^{\infty}\int_{\mathrm{TPR}^{-1}(\alpha)}^{\infty} p_+(s)p_-(t)dsdt \\ &= \int_{\mathrm{FPR}^{-1}(\beta)}^{\mathrm{FPR}^{-1}(\gamma)}\int_{\mathrm{TPR}^{-1}(\alpha)}^{\infty} p_+(s)p_-(t)dsdt.\end{aligned}$$

As a result,

$$\begin{aligned}&\mathrm{TPAUC}(h|\mathrm{TPR}\geq\alpha,\mathrm{FPR}\leq\beta) = \mathrm{OPAUC}(h|\mathrm{FPR}\in(\gamma,\beta)) - (\beta-\gamma)\alpha \\ &= \int_{\mathrm{FPR}^{-1}(\beta)}^{\mathrm{FPR}^{-1}(\gamma)}\int_{t}^{\infty} p_+(s)p_-(t)dsdt - \int_{\mathrm{FPR}^{-1}(\beta)}^{\mathrm{FPR}^{-1}(\gamma)}\int_{\mathrm{TPR}^{-1}(\alpha)}^{\infty} p_+(s)p_-(t)dsdt \\ &= \int_{\mathrm{FPR}^{-1}(\beta)}^{\mathrm{FPR}^{-1}(\gamma)}\int_{t}^{\mathrm{TPR}^{-1}(\alpha)} p_+(s)p_-(t)dsdt = \int_{\mathrm{FPR}^{-1}(\beta)}^{\infty}\int_{t}^{\mathrm{TPR}^{-1}(\alpha)} p_+(s)p_-(t)dsdt,\end{aligned}$$

where the last equality follows from $\mathrm{FPR}^{-1}(\gamma) = \mathrm{TPR}^{-1}(\alpha)$. Thus,

$$\begin{aligned}\mathrm{TPAUC}(h|\mathrm{TPR}\geq\alpha,\mathrm{FPR}\leq\beta) &= \int_{\mathrm{FPR}^{-1}(\beta)}^{\infty}\int_{t}^{\mathrm{TPR}^{-1}(\alpha)} p_+(s)p_-(t)dsdt \\ &= \int_{\mathrm{FPR}^{-1}(\beta)}^{\infty}\int_{-\infty}^{\mathrm{TPR}^{-1}(\alpha)} p_+(s)p_-(t)\mathbb{I}(s>t)dsdt.\end{aligned}$$

This concludes the proof of the second part. □

Hence, an empirical estimator of OPAUC with FPR restricted in the range $[\alpha,\beta]$ can be computed by

$$\frac{1}{n_+ k_1} \sum_{\mathbf{x}_i \in \mathcal{S}_+} \sum_{\mathbf{x}_j \in \mathcal{S}_-^{\downarrow}[k_1+1, k_2]} \mathbb{I}(h(\mathbf{x}_+) > h(\mathbf{x}_-)), \tag{2.39}$$

where $k_1 = \lceil n_- \alpha \rceil$, $k_2 = \lfloor n_- \beta \rfloor$, and $\mathcal{S}^{\downarrow}[k_1, k_2] \subseteq \mathcal{S}$ denotes the subset of examples whose rank in terms of their prediction scores in the descending order are in the range of $[k_1, k_2]$.

An empirical estimator of TPUC with with FPR restricted in a range of $[0, \beta]$ and TPR restricted in a range of $[\alpha, 1]$ is computed by:

$$\frac{1}{k_1} \frac{1}{k_2} \sum_{\mathbf{x}_i \in \mathcal{S}_+^{\uparrow}[1,k_1]} \sum_{\mathbf{x}_j \in \mathcal{S}_-^{\downarrow}[1,k_2]} \mathbb{I}(h(\mathbf{w}; \mathbf{x}_i) > h(\mathbf{w}; \mathbf{x}_i)), \tag{2.40}$$

where $k_1 = \lceil n_+(1-\alpha) \rceil$, $k_2 = \lfloor n_- \beta \rfloor$, and $\mathcal{S}^{\uparrow}[k_1, k_2] \subseteq \mathcal{S}$ denotes the subset of examples whose rank in terms of their prediction scores in the ascending order are in the range of $[k_1, k_2]$.

Necessity of Maximizing partial AUC

In many applications, there are large monetary costs due to high false positive rates (FPR) and low true positive rates (TPR), e.g., in medical diagnosis. Hence, a measure of interest would be the pAUC- the region of the ROC curve corresponding to low FPR and/or high TPR. With a similar argument as last section, a model that maximizes AUC does not necessarily optimizes pAUC. Let us compare two models on a dataset with 2 positive and 100 negative molecules (Figure 2.4). The model 1 ranks two negatives above the two positives followed by the remaining 98 negatives. The model 2 ranks one positive at the top, and then four negatives above the other positive followed by the remaining 96 negatives. The two models have the same AUC score of $196/200 = 0.98$ but have different pAUC scores. When restricting FPR $\in [0, 0.02]$, model 1 has an empirical pAUC score of $\frac{0}{4} = 0$ and model 2 has an empirical pAUC score of $\frac{2}{4} = 0.5$ according to (2.39).

> **Critical:** Partial AUC emphasize the correct order between the top ranked negative data and/or the bottom ranked positive data.

A Direct Formulation

Using a surrogate loss of zero-one loss, OPAUC maximization for learning a parameterized model $h(\mathbf{w}; \cdot)$ can be formulated as:

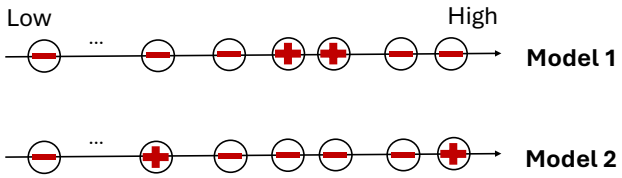


Fig. 2.4: Two models that have the same AUC score but differ dramatically in pAUC. The arrows indicate the prediction scores from low to high.

$$\min_{\mathbf{w}} \frac{1}{n_+}\frac{1}{k_2} \sum_{\mathbf{x}_i \in \mathcal{S}_+} \sum_{\mathbf{x}_j \in \mathcal{S}_-^{\downarrow}[1,k_2]} \ell(h(\mathbf{w};\mathbf{x}_j) - h(\mathbf{w};\mathbf{x}_i)). \tag{2.41}$$

Similarly, TPAUC maximization can be formulated as:

$$\min_{\mathbf{w}} \frac{1}{k_1}\frac{1}{k_2} \sum_{\mathbf{x}_i \in \mathcal{S}_+^{\uparrow}[1,k_1]} \sum_{\mathbf{x}_j \in \mathcal{S}_-^{\downarrow}[1,k_2]} \ell(h(\mathbf{w};\mathbf{x}_j) - h(\mathbf{w};\mathbf{x}_i)), \tag{2.42}$$

where $k_1 = \lfloor n_+(1-\alpha) \rfloor$, $k_2 = \lfloor n_-\beta \rfloor$.

Both problems are not standard ERM. The challenge for solving the above problems is that the selection of examples in a range, e.g., $\mathcal{S}_-^{\downarrow}[1,k_2]$ and $\mathcal{S}_+^{\uparrow}[1,k_1]$, is not only expensive but also non-differentiable. We will explore different approaches for optimizing OPAUC and TPUC in Section 6.4.3 using advanced compositional optimization techniques.

An Indirect Formulation

When the surrogate loss $\ell(t)$ is non-decreasing, the top-k selector of negative examples $\mathcal{S}_-^{\downarrow}[1,k_2]$ can be transferred into the top-k average of pairwise losses, which becomes an CVaR. By drawing the connection between CVaR and KL-regularized DRO, an indirect objective for OPAUC maximization is formulated by:

$$\min_{\mathbf{w}} \frac{1}{n_+} \sum_{\mathbf{x}_i \in \mathcal{S}_+} \tau \log\left(\sum_{\mathbf{x}_j \in \mathcal{S}_-} \exp\left(\frac{\ell(h(\mathbf{w};\mathbf{x}_j) - h(\mathbf{w};\mathbf{x}_i))}{\tau} \right) \right). \tag{2.43}$$

This problem is an instance of EXM, which will be solved by FCCO techniques. TPAUC maximization can be handled similarly. We will present detailed exposition in Chapter 6.4.3.

2.3.4 Ranking Losses

Ranking losses are commonly employed in learning to rank.

What is Learning to Rank?

Learning to rank (LTR) is a machine learning problem that aims to learn a ranking model, which can be used to predict the relevance order of a set of items given a query.

Let Q denote the query set of size N, and let $q \in Q$ represent an individual query. For each query q, let $\mathcal{S}_q$ be a set of N_q items (e.g., documents, movies) to be ranked. For each item $\mathbf{x}_{q,i} \in \mathcal{S}_q$, let $y_{q,i} \in \mathbb{R}^+$ denote its relevance score, which quantifies the relevance between the query q and the item $\mathbf{x}_{q,i}$. Define $\mathcal{S}_q^+ \subseteq \mathcal{S}_q$ as the subset of N_q^+ items relevant to q, i.e., those with non-zero relevance scores. Let $\mathcal{S} = \{(q, \mathbf{x}_{q,i}) \mid q \in Q, \mathbf{x}_{q,i} \in \mathcal{S}_q^+\}$ represent the collection of all relevant query-item (Q-I) pairs.

Let $s(\mathbf{w}; \mathbf{x}, q)$ denote the predicted relevance score for item $\mathbf{x}$ with respect to query q, parameterized by $\mathbf{w} \in \mathbb{R}^d$ (e.g., a deep neural network). Define the rank of item $\mathbf{x}$ within $\mathcal{S}_q$ as:

$$r(\mathbf{w}; \mathbf{x}, \mathcal{S}_q) = \sum_{\mathbf{x}' \in \mathcal{S}_q} \mathbb{I}(s(\mathbf{w}; \mathbf{x}', q) - s(\mathbf{w}; \mathbf{x}, q) \geq 0),$$

where ties are ignored.

NDCG and NDCG Loss

Normalized Discounted Cumulative Gain (NDCG) is a metric commonly used to evaluate the quality of ranking algorithms, especially in information retrieval and recommender systems.

NDCG evaluates how well a model ranks relevant items near the top of a list for a query q. The DCG of a ranked list according to $\{s(\mathbf{w}; \mathbf{x}, q), \mathbf{x} \in \mathcal{S}_q\}$ is given by:

$$\text{DCG}_q := \sum_{\mathbf{x} \in \mathcal{S}_q} \frac{2^{y_i} - 1}{\log_2(1 + r(\mathbf{w}; \mathbf{x}, \mathcal{S}_q))} = \sum_{\mathbf{x} \in \mathcal{S}_q^+} \frac{2^{y_i} - 1}{\log_2(1 + r(\mathbf{w}; \mathbf{x}, \mathcal{S}_q))}.$$

Note that the summation is over $\mathcal{S}_q^+$ rather than $\mathcal{S}_q$, as only relevant items contribute to the DCG score due to their non-zero relevance.

NDCG normalizes DCG by the ideal DCG denoted by Z_q of the best possible ranking:

$$\text{NDCG}_q = \frac{\text{DCG}_q}{Z_q}.$$

The average NDCG over all queries is given by:

$$\text{NDCG:} \quad \frac{1}{N}\sum_{q=1}^{N}\frac{1}{Z_q}\sum_{\mathbf{x}_{q,i}\in\mathcal{S}_q^+}\frac{2^{y_{q,i}}-1}{\log_2(r(\mathbf{w};\mathbf{x}_{q,i},\mathcal{S}_q)+1)}, \tag{2.44}$$

where Z_q can be precomputed.

By replacing the indicator function with a surrogate function in Table 2.3, we approximate $r(\mathbf{w};\mathbf{x},\mathcal{S}_q)/N_q$ by

$$g(\mathbf{w};\mathbf{x},\mathcal{S}_q) = \frac{1}{N_q}\sum_{\mathbf{x}'\in\mathcal{S}_q}\ell(s(\mathbf{w};\mathbf{x}',q)-s(\mathbf{w};\mathbf{x},q)).$$

Then the NDCG loss minimization is defined by

$$\min_{\mathbf{w}}\frac{1}{N}\sum_{q=1}^{N}\frac{1}{Z_q}\sum_{\mathbf{x}_{q,i}\in\mathcal{S}_q^+}\frac{1-2^{y_{q,i}}}{\log_2(N_q g(\mathbf{w};\mathbf{x}_{q,i},\mathcal{S}_q)+1)}, \tag{2.45}$$

which is an instance of EXM. We will explore FCCO techniques for solving this problem in Section 6.4.4.

Listwise Cross-Entropy Loss

Analogous to multi-class classification, we can define a listwise cross-entropy loss for ranking. This is based on modeling the probability that a specific item is ranked at the top:

$$P_{\text{top}}(\mathbf{x}\mid q) = \frac{\exp(s(\mathbf{w};\mathbf{x},q))}{\sum_{\mathbf{x}_j\in\mathcal{S}_q}\exp(s(\mathbf{w};\mathbf{x}_j,q))}. \tag{2.46}$$

Accordingly, the listwise cross-entropy loss for query q is defined as:

$$L(\mathbf{w};q) = \sum_{\mathbf{x}_{q,i}\in\mathcal{S}_q^+} -p_{q,i}\log\left(\frac{\exp(s(\mathbf{w};\mathbf{x}_{q,i},q))}{\sum_{\mathbf{x}_j\in\mathcal{S}_q}\exp(s(\mathbf{w};\mathbf{x}_j,q))}\right),$$

where $p_{q,i}$ denotes the top-one prior probability for item $\mathbf{x}_{q,i}$, such as

$$p_{q,i} = \frac{\exp(y_{q,i})}{\sum_{\mathbf{x}_{q,i}\in\mathcal{S}_q}\exp(y_{q,i})} \quad \text{or} \quad p_{q,i} = \frac{1}{N_q}.$$

An optimization objective based on the average of listwise cross-entropy losses over all queries leads to the following formulation known as ListNet:

$$\min_{\mathbf{w}} \quad \frac{1}{N}\sum_{q=1}^{N}\sum_{\mathbf{x}_{q,i}\in\mathcal{S}_q^+} p_{q,i}\log\left(\sum_{\mathbf{x}_j\in\mathcal{S}_q}\exp(s(\mathbf{w};\mathbf{x}_j,q)-s(\mathbf{w};\mathbf{x}_{q,i},q))\right). \tag{2.47}$$

This formulation closely resembles equation (2.43) and constitutes a special case of the EXM framework.

2.3.5 Contrastive Losses

Contrastive losses are commonly used in representation learning, which is a fundamental problem in the era of deep learning and modern AI.

What is Representation Learning?

Representation Learning is a process in machine learning where algorithms extract meaningful patterns from raw data (e.g., images) to create representations that are useful for many downstream tasks, e.g., learning a classifier or a retrieval model.

A deep neural network is usually used to extract representation from unstructured raw data. Let $h(\mathbf{w};\cdot) : \mathcal{X} \to \mathbb{R}^{d_1}$ denote the representation network that outputs an embedding vector, which is sometimes called the encoder. A meaningful encoder should capture the semantics such that 'similar' data points (positive pairs) are closer to each other and dissimilar data points (negative pairs) are far away from each other in the embedding space.

To conduct the representation learning, the following data is usually constructed. Let $\mathbf{x}_i$ be an anchor data, and let $\mathbf{x}_i^+$ denote a positive data of $\mathbf{x}_i$. Denote by $\mathcal{S}_i^-$ the set of negative data of $\mathbf{x}_i$. Let $s(\mathbf{w};\mathbf{x},\mathbf{y})$ denote a similarity score between the two encoded representations. For example, if $h(\mathbf{w};\mathbf{x})$ is a normalized vector such that $\|h(\mathbf{w};\mathbf{x})\|_2 = 1$, we can use $s(\mathbf{w};\mathbf{x},\mathbf{y}) = h(\mathbf{w};\mathbf{x})^\top h(\mathbf{w};\mathbf{y})$.

A contrastive loss for each positive pair $(\mathbf{x}_i, \mathbf{x}_i^+)$ is defined by:

$$L(\mathbf{w};\mathbf{x}_i,\mathbf{x}_i^+) = \tau \log\left(\frac{1}{|\mathcal{S}_i^-|}\sum_{\mathbf{y}\in\mathcal{S}_i^-}\exp((s(\mathbf{w};\mathbf{x}_i,\mathbf{y}) - s(\mathbf{w};\mathbf{x}_i,\mathbf{x}_i^+))/\tau)\right), \tag{2.48}$$

where $\tau > 0$ is called the temperature parameter. Given a set of data $\{(\mathbf{x}_i, \mathbf{x}_i^+, \mathcal{S}_i^-)\}_{i=1}^n$, minimizing a contrastive objective for representation learning is formulated as:

$$\min_{\mathbf{w}} \frac{1}{n}\sum_{i=1}^n \tau \log\left(\frac{1}{|\mathcal{S}_i^-|}\sum_{\mathbf{y}\in\mathcal{S}_i^-}\exp((s(\mathbf{w};\mathbf{x}_i,\mathbf{y}) - s(\mathbf{w};\mathbf{x}_i,\mathbf{x}_i^+))/\tau)\right). \tag{2.49}$$

Traditional supervised representation learning methods construct the positive and negative data using the annotated class labels, such that data in the same class are deemed as positive and data from different classes are considered as negative. However, this requires a large amount of labeled data to learn the encoder, which requires significant human effort in labeling. To address this issue, self-supervised represen-

tation learning (SSRL) techniques are employed to fully exploit the vast data readily available on the internet via self-supervision to learn representations that are useful for many downstream tasks. In SSRL, a positive pair $(\mathbf{x}_i, \mathbf{x}_i^+)$ may consist of different augmented views of the same sample or represent different modalities of the same underlying object (e.g., an image and its corresponding text). The negative samples for each anchor $\mathbf{x}_i$ are typically drawn from all other data points in the dataset excluding $\mathbf{x}_i$. In this setting, a variant of the contrastive objective is useful by adding a small constant $\varepsilon > 0$ inside the logarithm:

$$\min_{\mathbf{w}} \frac{1}{n} \sum_{i=1}^{n} \tau \log \left(\varepsilon + \frac{1}{|\mathcal{S}_i^-|} \sum_{\mathbf{y} \in \mathcal{S}_i^-} \exp((s(\mathbf{w}; \mathbf{x}_i, \mathbf{y}) - s(\mathbf{w}; \mathbf{x}_i, \mathbf{x}_i^+))/\tau) \right). \tag{2.50}$$

This can mitigate the impact of false negative data in $\mathcal{S}_i^-$. We will explore SSRL in Section 6.5.

Optimization Challenge

Optimizing the above contrastive objectives is challenging due to the presence of summations both inside and outside the logarithmic function. These losses can be reformulated as special cases of the X-risk, where the outer function is $f(g_i) = \tau \log(g_i)$, and g_i represents the inner average computed over negative samples associated with each $\mathbf{x}_i$.

2.4 Discriminative Data Prediction

The aforementioned X-risks can be unified under a principled discriminative learning framework for data prediction, providing a statistical foundation for developing advanced methods to train foundation models in modern AI.

What is a Foundation Model?

A foundation model (FM) is a type of machine learning model trained on large, diverse datasets (generally using self-supervision at scale) that can be adapted to a wide range of downstream tasks.

The widely used foundation models include Contrastive Language-image Pretrained (CLIP) model (see Section 6.5), Dense Passage Retrieval (DPR) model, large language models (LLMs) such as the Generative Pretrained Transformer (GPT) series (see Section 6.6), and vision-language models (VLMs). These models fall into two main categories: **representation models**, such as CLIP and DPR, and **generative models**, including LLMs and VLMs.

We present a discriminative data prediction framework to facilitate the learning of these foundation models. Suppose there exists a set of observed paired data, $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$, where $\mathbf{x}_i \in \mathcal{X}$ and $\mathbf{y}_i \in \mathcal{Y}$. These pairs typically represent real-world positive correspondences. While this setup resembles traditional supervised learning where $\mathbf{x}_i$ represents input data and $\mathbf{y}_i$ denotes a class label, there is a crucial difference: here, $\mathbf{y}_i$ refers to data from a **continuous space** (e.g., images) or an **uncountable space** (e.g., text). For instance:

- In training the CLIP model, $\mathbf{x}_i$ represents an image and $\mathbf{y}_i$ is the corresponding text caption (or vice versa).
- In training the DPR model, $\mathbf{x}_i$ is an input question, and $\mathbf{y}_i$ is the corresponding textual answer.
- In fine-tuning LLMs or VLMs, $\mathbf{x}_i$ represents input data (e.g., prompts or images), and $\mathbf{y}_i$ represents the text to be generated.

Discriminative Data Prediction

The problem of learning a representation model or fine-tuning a generative model can be framed as discriminative learning, which we term as data prediction, such that given any anchor data $\mathbf{x}$, the parameterized scoring function $s(\mathbf{w}; \cdot, \cdot)$ is able to discriminate a positive data $\mathbf{y}$ from any other negative data $\mathbf{y}'$, i.e., $s(\mathbf{w}; \mathbf{x}, \mathbf{y}) \geq s(\mathbf{w}; \mathbf{x}, \mathbf{y}')$.

Since the risk function usually involves coupling each positive data with many other possibly negative data points in a compositional structure, the resulting risk is called discriminative X-risk. The following subsections detail two specific approaches to formulating discriminative X-risks.

2.4.1 A Discriminative Probabilistic Modeling Approach

Without loss of generality, we assume that $\mathcal{X}$ and $\mathcal{Y}$ are continuous spaces. Let $\mathbb{P}_J$ denote the joint distribution of a pair $(\mathbf{x}, \mathbf{y})$, and let $\mathbb{P}_1$ and $\mathbb{P}_2$ denote the marginal distributions of $\mathbf{x}$ and $\mathbf{y}$, respectively. We write their corresponding density functions as $p(\cdot, \cdot)$, $p_1(\cdot)$, and $p_2(\cdot)$. We denote the conditional density functions by $p(\mathbf{y}|\mathbf{x})$ and $p(\mathbf{x}|\mathbf{y})$, corresponding to the conditional distributions $\mathbb{P}(\mathbf{y}|\mathbf{x})$ and $\mathbb{P}(\mathbf{x} \mid \mathbf{y})$. Below, we present two approaches based on discriminative probabilistic modeling (DPM)

Symmetric DPM

For symmetric DPM, we use $s(\mathbf{w}; \mathbf{x}, \mathbf{y})$ to model both conditional distributions $\mathbb{P}(\mathbf{y}|\mathbf{x})$ and $\mathbb{P}(\mathbf{x}|\mathbf{y})$. A discriminative probabilistic approach models the conditional probability $p(\mathbf{y}|\mathbf{x})$ using a scoring function $s(\mathbf{w}; \mathbf{x}, \mathbf{y})$ by:

$$p_{\mathbf{w}}(\mathbf{y}|\mathbf{x}) = \frac{p_2(\mathbf{y}) \exp(s(\mathbf{w}; \mathbf{x}, \mathbf{y})/\tau)}{\int_{\mathbf{y} \in \mathcal{Y}} p_2(\mathbf{y}) \exp(s(\mathbf{w}; \mathbf{x}, \mathbf{y}')/\tau) d\mathbf{y}'}, \tag{2.51}$$

where $\tau > 0$ is a temperature hyperparameter. The above parameterized distribution is the solution to the following problem for a fixed $\mathbf{x}$:

$$p_{\mathbf{w}}(\cdot|\mathbf{x}) = \arg\max_{\mathbb{Q} \in \mathcal{Q}} \mathbb{E}_{\mathbf{y}' \sim \mathbb{Q}} s(\mathbf{w}; \mathbf{x}, \mathbf{y}') - \tau \mathrm{KL}(\mathbb{Q}, \mathbb{P}_2),$$

where $\mathcal{Q} = \{\mathbb{Q} | \mathbb{Q} \ll \mathbb{P}_2\}$ is a set of probability distributions over $\mathbf{y} \in \mathcal{Y}$.

Similarly, we model $p(\mathbf{x}|\mathbf{y})$ as

$$p_{\mathbf{w}}(\mathbf{x}|\mathbf{y}) = \frac{p_1(\mathbf{x}) \exp(s(\mathbf{w}; \mathbf{x}, \mathbf{y})/\tau)}{\int_{\mathbf{x} \in \mathcal{X}} p_1(\mathbf{x}) \exp(s(\mathbf{w}; \mathbf{x}', \mathbf{y})/\tau) d\mathbf{x}'}. \tag{2.52}$$

Given a set of observed positive pairs $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$, the model parameters $\mathbf{w}$ are learned by minimizing the empirical risk of the negative log-likelihood:

$$\min_{\mathbf{w}} -\frac{1}{n} \sum_{i=1}^{n} \left\{ \tau \log \frac{\exp(s(\mathbf{w}; \mathbf{x}_i, \mathbf{y}_i)/\tau)}{\mathbb{E}_{\mathbf{y}' \sim \mathbb{P}_2} \exp(s(\mathbf{w}; \mathbf{x}_i, \mathbf{y}')/\tau)} + \tau \log \frac{\exp(s(\mathbf{w}; \mathbf{x}_i, \mathbf{y}_i)/\tau)}{\mathbb{E}_{\mathbf{x}' \sim \mathbb{P}_1} \exp(s(\mathbf{w}; \mathbf{x}', \mathbf{y}_i)/\tau)} \right\}.$$

A significant challenge in solving this problem lies in handling the partition functions,

$$Z(\mathbf{x}_i) = \mathbb{E}_{\mathbf{y}' \sim \mathbb{P}_2} \exp(s(\mathbf{w}; \mathbf{x}_i, \mathbf{y}')/\tau) d\mathbf{y}', \quad Z(\mathbf{y}_i) = \mathbb{E}_{\mathbf{x}' \sim \mathbb{P}_1} \exp(s(\mathbf{w}; \mathbf{x}', \mathbf{y}_i)/\tau),$$

which are often computationally intractable. To overcome this, an approximation can be constructed using a set of samples $\hat{\mathcal{Y}}_i \subseteq \mathcal{Y}$, $\hat{\mathcal{X}}_i \subseteq \mathcal{X}$. The partition functions are then estimated by:

$$\hat{Z}(\mathbf{x}_i) = \frac{1}{|\hat{\mathcal{Y}}_i|} \sum_{\hat{\mathbf{y}}_j \in \hat{\mathcal{Y}}_i} \exp(s(\mathbf{w}; \mathbf{x}_i, \hat{\mathbf{y}}_j)/\tau), \quad \hat{Z}(\mathbf{y}_i) = \frac{1}{|\hat{\mathcal{X}}_i|} \sum_{\hat{\mathbf{x}}_j \in \hat{\mathcal{X}}_i} \exp(s(\mathbf{w}; \hat{\mathbf{x}}_j, \mathbf{y}_i)/\tau).$$

Consequently, the resulting optimization problem is an empirical X-risk minimization problem:

$$\begin{aligned} \min_{\mathbf{w}} \frac{1}{n} \sum_{i=1}^{n} \tau \log \left(\sum_{\hat{y}_j \in \hat{\mathcal{Y}}_i} \exp\left(\frac{s(\mathbf{w}; \mathbf{x}_i, \hat{\mathbf{y}}_j) - s(\mathbf{w}; \mathbf{x}_i, \mathbf{y}_i)}{\tau} \right) \right) \\ + \tau \log \left(\sum_{\hat{x}_j \in \hat{\mathcal{X}}_i} \exp\left(\frac{s(\mathbf{w}; \hat{\mathbf{x}}_j, \mathbf{y}_i) - s(\mathbf{w}; \mathbf{x}_i, \mathbf{y}_i)}{\tau} \right) \right). \end{aligned} \tag{2.53}$$

The above approach can be justified that if $s(\mathbf{w}, \cdot, \cdot)$ is optimized over all possible scoring functions, then the learned $p_s(\mathbf{y}|\mathbf{x})$ and $p_s(\mathbf{x}|\mathbf{y})$ approaches the true density functions of $\mathbb{P}(\mathbf{y}|\mathbf{x})$ and $\mathbb{P}(\mathbf{x}|\mathbf{y})$ when n approaches ∞, respectively.

Theorem 2.4 *Let us consider the following problem over all possible scoring functions $s(\cdot,\cdot)$:*

$$\min_{s} -\mathbb{E}_{\mathbf{x},\mathbf{y}}\left[\tau \log \frac{p_2(\mathbf{y})\exp(s(\mathbf{x},\mathbf{y})/\tau)}{\mathbb{E}_{\mathbf{y}'\sim\mathbb{P}_2}\exp(s(\mathbf{x},\mathbf{y}')/\tau)} + \tau \log \frac{p_1(\mathbf{x})\exp(s(\mathbf{x},\mathbf{y})/\tau)}{\mathbb{E}_{\mathbf{x}'\sim\mathbb{P}_1}\exp(s(\mathbf{x}',\mathbf{y})/\tau)}\right]. \quad (2.54)$$

Then the set of global minimizers is given by

$$\mathcal{S}_* = \left\{ s : \frac{s(\mathbf{x},\mathbf{y})}{\tau} = \log \frac{p(\mathbf{x},\mathbf{y})}{p_1(\mathbf{x})p_2(\mathbf{y})} + const \right\},$$

where const is a constant, and we have

$$p_s(\mathbf{y}|\mathbf{x}) = \frac{p_2(\mathbf{y})\exp(s(\mathbf{x},\mathbf{y})/\tau)}{\int_{\mathbf{y}'\in\mathcal{Y}} p_2(\mathbf{y}')\exp(s(\mathbf{x},\mathbf{y}')/\tau)d\mathbf{y}'} = p(\mathbf{y}|\mathbf{x}),$$
$$p_s(\mathbf{x}|\mathbf{y}) = \frac{\mathbb{P}_1(\mathbf{y})\exp(s(\mathbf{x},\mathbf{y})/\tau)}{\int_{\mathbf{x}'\in\mathcal{X}} \mathbb{P}_1(\mathbf{x})\exp(s(\mathbf{x}',\mathbf{y})/\tau)d\mathbf{y}'} = p(\mathbf{x}|\mathbf{y}).$$

Proof. Let $\mathcal{F}_1$ be a class of functions $f_1(\mathbf{x},\mathbf{y}) : \mathcal{X}\times\mathcal{Y} \to \mathbb{R}$ such that $f_1(\mathbf{x},\mathbf{y}) \geq 0$ and $\int_{\mathbf{y}\in\mathcal{Y}} f_1(\mathbf{x},\mathbf{y}) = 1$, which induces a probability distribution $\mathbb{Q}_{1,x}(\cdot)$ over $\mathcal{Y}$ for any $\mathbf{x}$. Similarly, we define $f_2(\mathbf{x},\mathbf{y}) \in \mathcal{F}_2$ that induces a probability distribution $\mathbb{Q}_{2,y}(\cdot)$ over $\mathcal{X}$ for any $\mathbf{y}$.

Let us define a problem:

$$\min_{f_1\in\mathcal{F}_1, f_2\in\mathcal{F}_2} \mathbb{E}_{\mathbf{x},\mathbf{y}}[-\log f_1(\mathbf{x},\mathbf{y}) - \log f_2(\mathbf{x},\mathbf{y})].$$

Since

$$\begin{aligned}
&\mathbb{E}_{\mathbf{x},\mathbf{y}}[-\log f_1(\mathbf{x},\mathbf{y}) - \log f_2(\mathbf{x},\mathbf{y})] \\
&= \mathbb{E}_{\mathbf{x}}\mathbb{E}_{\mathbf{y}\sim\mathbb{P}(\cdot|\mathbf{x})}\left[-\log\frac{f_1(\mathbf{x},\mathbf{y})}{p(\mathbf{y}|\mathbf{x})} - \log p(\mathbf{y}|\mathbf{x})\right] \\
&+ \mathbb{E}_{\mathbf{y}}\mathbb{E}_{\mathbf{x}\sim\mathbb{P}(\cdot|\mathbf{y})}\left[-\log\frac{f_2(\mathbf{x},\mathbf{y})}{p(\mathbf{x}|\mathbf{y})} - \log p(\mathbf{y}|\mathbf{x})\right] \\
&= \mathbb{E}_{\mathbf{x}}[\mathrm{KL}(\mathbb{P}(\cdot|\mathbf{x}),\mathbb{Q}_{1,x}(\cdot))] + \mathbb{E}_{\mathbf{y}}[\mathrm{KL}(\mathbb{P}(\cdot|\mathbf{y}),\mathbb{Q}_{2,y}(\cdot))] + \text{const},
\end{aligned}$$

where const is independent of f. Hence the minimizer $f_1^*(\mathbf{x},\mathbf{y})$ is equal to $p(\mathbf{y}|\mathbf{x})$ and the minimizer $f_2^*(\mathbf{x},\mathbf{y})$ is equal to $p(\mathbf{x}|\mathbf{y})$. As a result, for optimal $s_*(\cdot,\cdot)$ we require

$$\frac{p_2(\mathbf{y})\exp(s_*(\mathbf{x},\mathbf{y})/\tau)}{\int_{\mathcal{Y}} p_2(\mathbf{y})\exp(s_*(\mathbf{x},\mathbf{y}')/\tau)d\mathbf{y}'} = f_1^*(\mathbf{x},\mathbf{y}) = p(\mathbf{y}|\mathbf{x}), \quad (2.55)$$

$$\frac{p_1(\mathbf{x})\exp(s_*(\mathbf{x},\mathbf{y})/\tau)}{\int_{\mathcal{Y}} p_1(\mathbf{x})\exp(s_*(\mathbf{x}',\mathbf{y})/\tau)d\mathbf{x}'} = f_2^*(\mathbf{x},\mathbf{y}) = p(\mathbf{x}|\mathbf{y}). \quad (2.56)$$

From the first equation, we can derive that $s_*(\mathbf{x}, \mathbf{y}) = \log \frac{p(\mathbf{y}|\mathbf{x})}{p_2(\mathbf{y})} + h_1(\mathbf{x})$, where $h_1(\mathbf{x})$ is any arbitrary function of $\mathbf{x}$. From the second equation, we can derive that $s_*(\mathbf{x}, \mathbf{y}) = \log \frac{p(\mathbf{x}|\mathbf{y})}{p_1(\mathbf{x})} + h_2(\mathbf{x})$, where $h_2(\mathbf{y})$ is any arbitrary function of $\mathbf{y}$. As a result, the global minimizer $s_*(\mathbf{x}, \mathbf{y})$ will be in the form of $\log \frac{p(\mathbf{x},\mathbf{y})}{p_1(\mathbf{x})p_2(\mathbf{y})} + \text{const}$. □

One-sided DPM

If we are only interested in modeling $\mathbb{P}(\mathbf{y}|\mathbf{x})$, then we can consider one-sided DPM. We define the following parametric probability function to model $\mathbb{P}(\mathbf{y}|\mathbf{x})$:

$$p_{\mathbf{w}}(\mathbf{y}|\mathbf{x}) = \frac{\exp(s(\mathbf{w}; \mathbf{x}, \mathbf{y})/\tau)}{\int_{\mathcal{Y}} \exp(s(\mathbf{w}; \mathbf{x}, \mathbf{y}')/\tau) d\mu(\mathbf{y}')}, \tag{2.57}$$

where $\tau > 0$ is a temperature hyperparameter, and μ is the Lebesgue measure associated with the space $\mathcal{Y}$.

Given a set of observed positive pairs $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$, the model parameters $\mathbf{w}$ are learned by minimizing the empirical risk of the negative log-likelihood:

$$\min_{\mathbf{w}} -\frac{1}{n} \sum_{i=1}^{n} \tau \log \frac{\exp(s(\mathbf{w}; \mathbf{x}_i, \mathbf{y}_i)/\tau)}{\int_{\mathcal{Y}} \exp(s(\mathbf{w}; \mathbf{x}_i, \mathbf{y}')/\tau) d\mu(\mathbf{y}')}.$$

A significant challenge in solving this problem lies in handling the partition function,

$$Z_i = \int_{\mathcal{Y}} \exp(s(\mathbf{w}; \mathbf{x}_i, \mathbf{y}')/\tau) d\mu(\mathbf{y}'),$$

which is often computationally intractable. To overcome this, an approximation can be constructed using a set of samples $\hat{\mathcal{Y}}_i \subseteq \mathcal{Y}$. The partition function is then estimated as:

$$\hat{Z}_i = \sum_{\hat{\mathbf{y}}_j \in \hat{\mathcal{Y}}_i} \frac{1}{q_j} \exp(s(\mathbf{w}; \mathbf{x}_i, \hat{\mathbf{y}}_j)/\tau),$$

where q_j is an importance weight that accounts for the sample probability of $\hat{\mathbf{y}}_j$. Consequently, the empirical X-risk minimization problem is reformulated as:

$$\min_{\mathbf{w}} \frac{1}{n} \sum_{i=1}^{n} \tau \log \left(\sum_{\hat{\mathbf{y}}_j \in \hat{\mathcal{Y}}_i} \exp((s(\mathbf{w}; \mathbf{x}_i, \hat{\mathbf{y}}_j) + \zeta_j - s(\mathbf{w}; \mathbf{x}_i, \mathbf{y}_i))/\tau) \right),$$

where $\zeta_j = \tau \ln \frac{1}{q_j}$.

We can similarly justify the above approach by the following theorem.

Theorem 2.5 *Let us consider the following problem over all possible scoring functions $s(\cdot, \cdot)$:*

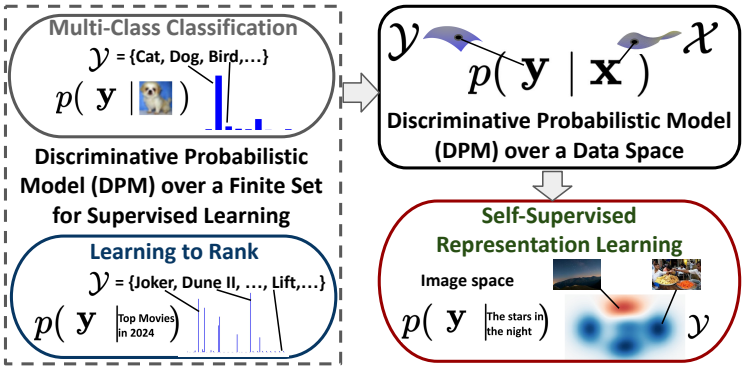


Fig. 2.5: DPM for supervised learning and self-supervised representation learning.

$$\min_{s} -\mathbb{E}_{\mathbf{x},\mathbf{y}} \tau \log \frac{\exp(s(\mathbf{x},\mathbf{y})/\tau)}{\int_{\mathbf{y}'\in\mathcal{Y}} \exp(s(\mathbf{x},\mathbf{y}')/\tau) d\mu(\mathbf{y}')}. \tag{2.58}$$

Then the set of global minimizers is given by

$$\mathcal{S}_* = \left\{ s : \frac{s(\mathbf{x},\mathbf{y})}{\tau} = \log p(\mathbf{y}|\mathbf{x}) + h(\mathbf{x}) \right\},$$

where $h(\cdot)$ *is an arbitrary function of* $\mathbf{x}$*, and we have* $p_s(\mathbf{y}|\mathbf{x}) = \frac{\exp(s(\mathbf{x},\mathbf{y})/\tau)}{\int_{\mathcal{Y}} \exp(s(\mathbf{x},\mathbf{y}')/\tau) d\mathbf{y}'} = p(\mathbf{y}|\mathbf{x})$.

The proof is similar to the previous one and thus is omitted.

Instantiation

The fundamental difference between symmetric DPM and one-sided DPM lies in what their scoring functions $s(\mathbf{w};\mathbf{x},\mathbf{y})$ are designed to capture. We can use symmetric DPM for learning *representation models* and one-sided DPM for learning generative models and supervised prediction models.

The standard cross-entropy loss for classification and the listwise cross-entropy loss for learning to rank can both be viewed as special cases of the one-sided DPM framework, where $\mathcal{Y}$ represents either a finite set of class labels or a list of items to be ranked for each query. In these cases, the integral naturally simplifies to a finite summation, eliminating the need to approximate the normalization term Z_i. However, when $\mathcal{Y}$ is large, computing Z_i remains computationally demanding. This challenge, in turn, motivates the development of more advanced compositional optimization techniques.

For representation learning, the goal is to learn a symmetric scoring function $s(\mathbf{w};\mathbf{x},\mathbf{y}) = h_1(\mathbf{w};\mathbf{x})^\top h_2(\mathbf{w};\mathbf{y})$ that approximates the global optimum

$$s^*(\mathbf{x}, \mathbf{y}) = \tau \log \frac{p(\mathbf{x}, \mathbf{y})}{p_1(\mathbf{x}) p_2(\mathbf{y})} + \text{const},$$

which measures how much the joint distribution $\mathbb{P}(\mathbf{x}, \mathbf{y})$ deviates from independence between $\mathbf{x}$ and $\mathbf{y}$. We will consider contrastive losses of CLIP in Section 6.5 for multi-modal representation learning, which can be interpreted by the symmetric DPM with $\mathbf{x}, \mathbf{y}$ denoting an image-text pair.

For generative modeling, we can use underlying models to induce a scoring function $s(\mathbf{w}; \mathbf{x}, \mathbf{y})$ for approximating the global optimum $s^*(\mathbf{x}, \mathbf{y}) = \tau \log p(\mathbf{y}|\mathbf{x}) + h(\mathbf{x})$. We will also consider discriminative fine-tuning of LLMs Section 6.6, which can be interpreted by the one-sided DPM with $\mathbf{x}, \mathbf{y}$ denoting an input-output pair.

An illustration of the connection between the probabilistic model for multi-modal representation learning and traditional supervised learning tasks including multi-class classification and learning to rank is shown in Figure 2.5.

Critical: Discriminative probabilistic model over a data space is a framework that unifies traditional label prediction and data ranking of supervised learning and modern self-supervised representation learning, and induces new approaches for fine-tuning LLMs.

2.4.2 A Robust Optimization Approach

The goal of discriminative learning is to increase the score $s(\mathbf{w}; \mathbf{x}, \mathbf{y}_+)$ for a "positive" pair $(\mathbf{x}, \mathbf{y}_+)$ while decreasing the score $s(\mathbf{w}; \mathbf{x}, \mathbf{y}_-)$ for any "negative" pair $(\mathbf{x}, \mathbf{y}_-)$.

Full Supervised setting

Let us first consider the supervised learning setting, where positive and negative samples are labeled, i.e., there is a function $r(\mathbf{x}, \mathbf{y}) \in (0, 1)$ that indicates whether they form a positive pair or a negative pair. We let $(\mathbf{x}, \mathbf{y}_+) \sim \mathbb{P}_+(\mathbf{x}, \mathbf{y}_+)$ denote a positive pair and $(\mathbf{x}, \mathbf{y}_-) \sim \mathbb{P}_-(\mathbf{x}, \mathbf{y}_-)$ denote a negative pair, where $\mathbb{P}_+(\mathbf{x}, \mathbf{y}_+) = \mathbb{P}(\mathbf{x})\mathbb{P}_+(\mathbf{y}_+|\mathbf{x}), \mathbb{P}_-(\mathbf{x}, \mathbf{y}_-) = \mathbb{P}(\mathbf{x})\mathbb{P}_-(\mathbf{y}_-|\mathbf{x})$, and $\mathbb{P}(\mathbf{x}, \mathbf{y}_+, \mathbf{y}_-) = \mathbb{P}_+(\mathbf{y}_+|\mathbf{x})\mathbb{P}_-(\mathbf{y}_-|\mathbf{x})\mathbb{P}(\mathbf{x})$. Let us denote a pairwise loss by $\ell(s(\mathbf{w}; \mathbf{x}, \mathbf{y}_-) - s(\mathbf{w}; \mathbf{x}, \mathbf{y}_+))$.

A naive goal is to minimize the expected risk:

$$\min_{\mathbf{w}} \mathbb{E}_{\mathbf{x}, \mathbf{y}_+, \mathbf{y}_- \sim \mathbb{P}(\mathbf{x}, \mathbf{y}_+, \mathbf{y}_-)} \left[\ell(s(\mathbf{w}; \mathbf{x}, \mathbf{y}_-) - s(\mathbf{w}; \mathbf{x}, \mathbf{y}_+)) \right].$$

However, a fundamental challenge for data prediction is that the number of negative data is usually much larger than the number of positive data. Hence, the expected risk is not a strong measure. To address this challenge, we can leverage OCE. In particular, we replace the expected risk $\mathbb{E}_{\mathbf{y}_- \sim \mathbb{P}(\mathbf{y}_-|\mathbf{x})} \left[\ell(s(\mathbf{w}; \mathbf{x}, \mathbf{y}_-) - s(\mathbf{w}; \mathbf{x}, \mathbf{y}_+)) \right]$ by its OCE counterpart, resulting the following population risk:

$$\min_{\mathbf{w}} \mathbb{E}_{\mathbf{x},\mathbf{y}_+}\left[\min_{\nu} \tau \mathbb{E}_{\mathbf{y}_-|\mathbf{x}} \phi^*\left(\frac{\ell(s(\mathbf{w};\mathbf{x},\mathbf{y}_-) - s(\mathbf{w};\mathbf{x},\mathbf{y}_+)) - \nu}{\tau}\right) + \nu\right]. \tag{2.59}$$

If the training dataset is $\mathcal{S} = \{\mathbf{x}_i, \mathbf{y}_i^+, \mathbf{y}_{ij}^-, i \in [n], j \in [m]\}$, where $\mathbf{y}_i^+ \sim \mathbb{P}_+(\cdot|\mathbf{x}_i)$ and $\mathbf{y}_{ij}^- \sim \mathbb{P}_-(\cdot|\mathbf{x}_i)$, then the empirical version becomes:

$$\min_{\mathbf{w}} \frac{1}{n}\sum_{i=1}^n \min_{\nu_i} \tau \frac{1}{m}\sum_{j=1}^m \phi^*\left(\frac{\ell(s(\mathbf{w};\mathbf{x}_i,\mathbf{y}_{ij}^-) - s(\mathbf{w};\mathbf{x}_i,\mathbf{y}_i^+)) - \nu_i}{\tau}\right) + \nu_i. \tag{2.60}$$

Semi-supervised setting

We can extend the above framework to the semi-supervised learning setting, where we only have samples from the positive distribution $\mathbb{P}_+(\cdot|\mathbf{x})$ and samples from the distribution $P(\cdot|\mathbf{x})$.

Let us assume that $\mathbb{P}(\cdot|\mathbf{x}) = \pi_+(\mathbf{x})\mathbb{P}_+(\cdot|\mathbf{x}) + \pi_-(\mathbf{x})\mathbb{P}_-(\cdot|\mathbf{x})$ and $\pi_+(\mathbf{x}) \ll \pi_-(\mathbf{x})$. This means that for a fixed data $\mathbf{x}$, the sampled data $\mathbf{y} \sim P(\cdot|\mathbf{x})$ is mostly likely from the negative distribution $\mathbb{P}_-(\cdot|\mathbf{x})$. Hence, we can approximate $\mathbb{E}_{\mathbf{y}_-\sim\mathbb{P}_-(\cdot|\mathbf{x})}$ by $\mathbb{E}_{\mathbf{y}\sim\mathbb{P}(\cdot|\mathbf{x})}$. Hence, a population risk in the semi-supervised learning setting becomes

$$\min_{\mathbf{w}} \mathbb{E}_{\mathbf{x},\mathbf{y}_+}\left[\min_{\nu} \tau \mathbb{E}_{\mathbf{y}|\mathbf{x}} \phi^*\left(\frac{\ell(s(\mathbf{w};\mathbf{x},\mathbf{y}) - s(\mathbf{w};\mathbf{x},\mathbf{y}_+)) - \nu}{\tau}\right) + \nu\right], \tag{2.61}$$

and its empirical version becomes

$$\min_{\mathbf{w}} \frac{1}{n}\sum_{i=1}^n \min_{\nu_i} \tau \frac{1}{m}\sum_{j=1}^m \phi^*\left(\frac{\ell(s(\mathbf{w};\mathbf{x}_i,\mathbf{y}_{ij}) - s(\mathbf{w};\mathbf{x}_i,\mathbf{y}_i^+)) - \nu_i}{\tau}\right) + \nu_i, \tag{2.62}$$

where $\{\mathbf{y}_{ij}, j = 1,\ldots,m\}$ are samples from $\mathbb{P}(\cdot|\mathbf{x})$.

Self-supervised setting

For self-supervised learning, we let $(\mathbf{x},\mathbf{y}^+) \sim \mathbb{P}(\mathbf{x},\mathbf{y}^+)$ denote a "positive" pair, and $(\mathbf{x},\mathbf{y}^-) \sim \mathbb{P}(\mathbf{x})\mathbb{P}(\mathbf{y}^-)$ denote a "negative" pair. For empirical learning, we only have a training set of $\mathcal{S} = \{\mathbf{x}_i, \mathbf{y}_i^+, i \in [n]\}$. We use $\mathcal{S}_i^- = \{\mathbf{y}_j^+\}_{j\neq i}$ to define the empirical risk:

$$\min_{\mathbf{w}} \frac{1}{n}\sum_{i=1}^n \min_{\nu_i} \tau \frac{1}{|\mathcal{S}_i^-|}\sum_{\mathbf{y}'\in\mathcal{S}_i^-} \phi^*\left(\frac{\ell(s(\mathbf{w};\mathbf{x}_i,\mathbf{y}') - s(\mathbf{w};\mathbf{x}_i,\mathbf{y}_i^+)) - \nu_i}{\tau}\right) + \nu_i. \tag{2.63}$$

We refer to the problems in (2.60), (2.62) and (2.63) as the Compositional OCE (COCE) optimization. We will present and analyze stochastic algorithms for solving COCE optimization in Chapter 5[Section 5.5].

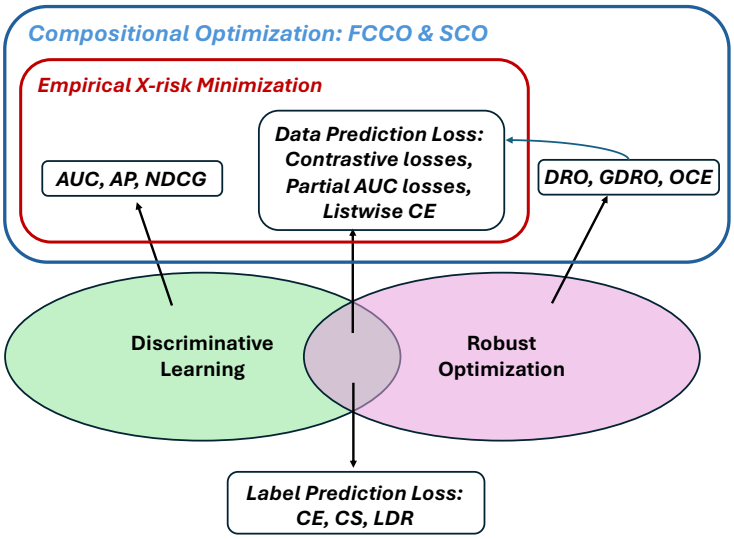


Fig. 2.6: Overview of different losses and two fundamental learning principles

Instantiation

When $\phi(t) = t\log t - t + 1$, the inner optimization over v_i in (2.62) admits a closed-form solution, which can be substituted back into the objective, yielding:

$$\min_{\mathbf{w}} \frac{1}{n}\sum_{i=1}^{n} \tau \log\left(\frac{1}{m}\sum_{j=1}^{m} \exp\left(\frac{\ell(s(\mathbf{w};\mathbf{x}_i,\mathbf{y}_{ij}) - s(\mathbf{w};\mathbf{x}_i,\mathbf{y}_i^+))}{\tau}\right)\right). \tag{2.64}$$

This formulation unifies several well-known losses as special cases:

- **Cross-Entropy Loss for Classification**: Let $\mathbf{x}_i$ denote an input data point, let y_i^+ represent its true class label and $\{y_{ij}, j = 1,\ldots,m\} = \{1,\ldots,K\}$ forms the full label space. Define the prediction score for the y-th class of $\mathbf{x}$ as $s(\mathbf{w};\mathbf{x},y) = h_0(\mathbf{w}_0;\mathbf{x})^\top \mathbf{w}_y$. When the loss function is $\ell(s) = s$ and $\tau = 1$, the objective reduces to the empirical risk with the standard cross-entropy loss.
- **Listwise Cross-Entropy Loss for Ranking**: Let $\mathbf{x}_i$ denote a query, $\{\mathbf{y}_i^+\}$ denote a relevant (positive) document, and $\{\mathbf{y}_{ij}\}_{j=1}^m$ denote the complete candidate list to be ranked. Let $s(\mathbf{w};\mathbf{x},\mathbf{y})$ be the predicted relevance score between a query $\mathbf{x}$ and a document $\mathbf{y}$. When the loss function is $\ell(s) = s$ and $\tau = 1$, the objective simplifies to the listwise cross-entropy loss.
- **Self-supervised Contrastive Loss for Representation Learning**: If $\mathbf{x}_i$ is an anchor (e.g., an image), $\mathbf{y}_i^+$ denotes its positive pair (e.g., the corresponding text) and $\{\mathbf{y}_{i,j}, j = 1,\ldots,m\} = \mathcal{S}_i^-$, the the objective in (2.64) recovers the contrastive loss (2.48) used in self-supervised contrastive representation learning.

- **Partial AUC Loss for Imbalanced Binary Classification**: Let $\mathbf{x}_i$ be a fixed class label ($i = 1$), with $\{\mathbf{y}_i^+\}$ denoting its positive data set and $\{\mathbf{y}_{ij}\}_{j=1}^m$ being its negative data set. Define the scoring function as $s(\mathbf{w}; \mathbf{x}, \mathbf{y}) = h(\mathbf{w}; \mathbf{y}) \in \mathbb{R}$. Under this setting, the objective in (2.64) reduces to the partial AUC loss in (2.43).

This framework offers a flexible foundation for designing alternative contrastive objectives by varying the loss function $\ell(\cdot)$, the temperature τ, the divergence function $\phi(\cdot)$, and the distributionally robust optimization (DRO) formulation, including its constrained variants.

Finally, Figure 2.6 illustrates the losses, objectives, and learning frameworks discussed in this chapter, along with their connections to the principles of discriminative learning and robust optimization. This perspective highlights the necessity of stochastic compositional optimization and finite-sum coupled compositional optimization, which will be presented in subsequent chapters.

2.5 History and Notes

Loss functions

A pioneering work analyzing the infinite-sample consistency of various multi-class surrogate loss functions is provided by Zhang (2004b). This work proves the consistency of several losses, including the cross-entropy loss. It also shows that the consistency of the Crammer-Singer and hinge losses can fail unless the maximum conditional probability of a class label given the input exceeds 0.5.

The Label-Distribution-Aware Margin (LDAM) Loss was proposed and studied by Cao et al. (2019), inspired by margin-based generalization error bounds tailored for each class. The label distributionally robust (LDR) losses and their consistency was proposed and studied by Zhu et al. (2023b).

Variants of standard loss functions have been developed to minimize the top-k error for $k > 1$, such as the top-k SVM loss and the top-k cross-entropy loss (Lapin et al., 2018; Yang and Koyejo, 2020). The top-k SVM loss can be recovered as a special case of the general LDR loss by setting $R(\mathbf{p}) = 0$ and $\Omega = \{\mathbf{p} \in \Delta_K : p_k \leq 1/k\}$. Although this formulation is generally inconsistent, adding a small strongly convex regularizer $R(\mathbf{p})$ to the LDR loss can restore consistency.

A sufficient condition for a loss function to be noise-tolerant is the symmetry property, as introduced by Ghosh et al. (2017). A loss function is considered noise-tolerant if the minimizer of the expected risk under the true label distribution remains the same under the noisy label distribution, provided the noise level is not excessively high.

Robust optimization

Robust optimization dates back to Scarf (1958), who studied an inventory problem in which the goal is to determine the purchase quantity that maximizes profit when future demand is a random variable whose underlying probability distribution is assumed to belong to a set of plausible distributions. The problem is reformulated as a worst-case analysis over all distributions in this set with known mean and variance. Later, Dupačová (1966) investigated the min−max robust formulation of stochastic linear programming. Since then, robust optimization has been extensively studied in management science, operations research, and mathematical programming (Kouvelis and Yu, 1997; Shapiro and Kleywegt, 2002; Rustem and Howe, 2002; Ben-Tal et al., 2009b). The term *distributionally robust optimization* was introduced by Delage and Ye (2010).

The ϕ-divergence (sometimes called f-divergence, where both f and ϕ denote a function) was introduced by Csiszár (1967). The use of ϕ-divergence to define the uncertainty set in robust optimization was first studied by Ben-Tal et al. (2013), while earlier works had considered using the KL divergence to define an uncertainty set of probabilities (Calafiore, 2007). A special case of DRO, namely the maximal loss, was shown to be beneficial for imbalanced classification by Shalev-Shwartz and Wexler (2016). The popularity of DRO in machine learning is largely attributed to Namkoong and Duchi (2017), who established a variance-based generalization error bound for DRO with the χ^2 divergence, building on their preceding work (Duchi et al., 2022). The optimized certainty equivalent (OCE) was proposed by Ben-Tal and Teboulle (1986b), and its connection to DRO was later established in (Ben-Tal and Teboulle, 2007). Group DRO was first proposed by Hu et al. (2018) and became widely recognized due to Sagawa et al. (2019).

AUC and NDCG

The receiver operating characteristic (ROC) curve was originally developed in the 1940s by electrical and radar engineers during World War II to detect enemy objects on the battlefield, which gave rise to its name (”receiver operating characteristic”) (Marcum, 1947). It was subsequently formalized within the framework of signal detection theory (Green and Swets, 1966). The probabilistic interpretation of AUC and its equivalence to the Mann−Whitney U-statistic (or Wilcoxon statistic) were later established by Hanley and McNeil (1982). The concept was subsequently introduced into machine learning as a standard metric for evaluating learning algorithms (Spackman, 1989). The first study of the one-way partial AUC (pAUC) was presented by Dodd and Pepe (2003), and the notion of two-way partial AUC was later introduced by Yang et al. (2019).

The study of AUC maximization dates back to Verrelst et al. (1998) and has since been extensively explored in machine learning. Yan et al. (2003) were the first to apply the gradient descent method to optimize a hinge-based pairwise surrogate loss for AUC, while Cortes and Mohri (2003) employed the Rankboost algorithm (Freund

et al., 2003) to optimize AUC. The compositional objective for AUC maximization was first proposed by Ying et al. (2016a) in a min–max form and was later generalized in (Yuan et al., 2021; Zhu et al., 2022c). For a comprehensive overview of related work, see the survey by Yang and Ying (2022). The first work on maximizing average precision was conducted by Morgan et al. (2004). The use of DRO for formulating partial AUC losses was proposed by Zhu et al. (2022a).

NDCG was introduced by Järvelin and Kekäläinen (2000), and the listwise cross-entropy loss for learning to rank was proposed by Cao et al. (2007). The concept of empirical X-risk minimization for unifying a family of non-decomposable losses was developed by the author of this book in (Yang, 2022), which also presents additional examples of X-risks.

Foundation Models

Representation learning in traditional machine learning is related to principal component analysis and distance metric learning (Yang and Jin, 2006). Conventional contrastive losses are defined on pairs $(\mathbf{x}, \mathbf{y})$ using it binary label indicating positive or negative pair (Hadsell et al., 2006) or triplets $(\mathbf{x}, \mathbf{y}_+, \mathbf{y}_-)$ (Weinberger and Saul, 2009). The Contrastive loss defined on a list of negative data for a positive pair was first introduced by Sohn (2016).

The term foundation model was introduced by Bommasani et al. (2021). The use of DRO to formulate the contrastive loss was first proposed by Qiu et al. (2023), providing a principled approach for optimizing individualized temperature parameters. The discriminative probabilistic modeling approach for self-supervised representation learning was first explored by Wang et al. (2025).

Generalization Error

Generalization error analysis is a central topic in several classical machine learning texts (Shalev-Shwartz and Ben-David, 2014; Mohri et al., 2018) and in the statistical learning theory literature (Koltchinskii, 2011). Typically, uniform convergence bounds of the form $\sup_{\mathbf{w}\in\mathcal{W}} |\mathcal{R}(\mathbf{w}) - \mathcal{R}_S(\mathbf{w})|$ are derived using concentration inequalities, with dependencies on both the number of training samples n and the complexity of the hypothesis class. More recently, there has been growing interest in directly analyzing the generalization performance of models returned by stochastic optimization algorithms using stability-based techniques (Hardt et al., 2016; Lei and Ying, 2019).

Generalization error analyses for DRO and OCE objectives have been extensively developed in the literature: Brown (2007) established theoretical bounds for CVaR, Namkoong and Duchi (2017) developed bounds for χ^2-constrained DRO, and Lee et al. (2020) explored generalization for general OCE risk. However, the generalization error for compositional OCE is under-development.

Machine Learning texts

There are excellent textbooks on machine learning (Shalev-Shwartz and Ben-David, 2014; Mohri et al., 2018; Bishop, 2006) and on robust optimization (Ben-Tal et al., 2009a). However, to the best of our knowledge, this book is the first to provide a comprehensive and unified treatment of diverse loss functions and objectives, ranging from the traditional cross-entropy loss to the contrastive loss used in self-supervised representation learning, through the lens of robust optimization and discriminative learning.

Chapter 3
Classic: Stochastic Optimization

Abstract In this chapter, we introduce standard stochastic optimization problems and present key stochastic optimization algorithms along with their complexity analysis. While many important stochastic algorithms have been proposed for solving stochastic optimization and empirical risk minimization problems, we focus on seven foundational methods that gained prominence before the deep learning era. These algorithms have had a profound impact on machine learning and provide essential insights for understanding more advanced methods presented in later chapters. The selected algorithms include stochastic gradient descent (SGD), stochastic proximal gradient descent, stochastic mirror descent, adaptive gradient methods, stochastic coordinate descent, stochastic gradient descent ascent, and stochastic optimistic mirror prox. We concentrate on the complexity analysis in the convex setting.

Stochastic optimization is classical wisdom in motion!

Contents

Algorithm 1 `SGD`

1: **Input:** learning rate schedule $\{\eta_t\}_{t=1}^T$, starting point $\mathbf{w}_1$
2: **for** $t = 1, \ldots, T$ **do**
3: Compute an unbiased gradient estimator $\mathbf{z}_t = \nabla g(\mathbf{w}_t; \zeta_t)$
4: Update the model $\mathbf{w}$ by $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{z}_t$
5: **end for**

3.1 Stochastic Gradient Descent

Let us consider the following standard stochastic optimization problem:

$$\min_{\mathbf{w}} g(\mathbf{w}) := \mathbb{E}_\zeta [g(\mathbf{w}; \zeta)]. \tag{3.1}$$

If g is differentiable, the stochastic gradient descent (SGD) method takes the following update:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \nabla g(\mathbf{w}_t; \zeta_t), \tag{3.2}$$

where ζ_t is a random sample. If g is non-differentiable, we use a stochastic subgradient $\mathcal{G}(\mathbf{w}; \zeta)$ to update the model parameter:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathcal{G}(\mathbf{w}_t; \zeta_t). \tag{3.3}$$

The key assumption regarding the stochastic gradient or subgradient is the following.

Assumption 3.1. *For any* $\mathbf{w}$*, we have* $\mathbb{E}_\zeta [\nabla g(\mathbf{w}; \zeta)] = \nabla g(\mathbf{w})$ *or* $\mathbb{E}_\zeta [\mathcal{G}(\mathbf{w}; \zeta)] \in \partial g(\mathbf{w})$.

Explanation of SGD update

The update (3.2) is equivalent to:

$$\mathbf{w}_{t+1} = \arg\min_{\mathbf{w}} g(\mathbf{w}_t; \zeta_t) + \nabla g(\mathbf{w}_t; \zeta_t)^\top (\mathbf{w} - \mathbf{w}_t) + \frac{1}{2\eta_t} \|\mathbf{w} - \mathbf{w}_t\|_2^2. \tag{3.4}$$

The stochastic linear approximation $\tilde{g}(\mathbf{w}; \zeta_t) = g(\mathbf{w}_t; \zeta_t) + \nabla g(\mathbf{w}_t; \zeta_t)^\top (\mathbf{w} - \mathbf{w}_t)$ serves as a stochastic surrogate for $g(\mathbf{w})$. Since it is only an approximation, we avoid optimizing it directly; instead, we seek a solution close to $\mathbf{w}_t$ that minimizes this surrogate.

When SGD is applied to solving ERM (2.1), ζ_t could represent one randomly sampled data with an index from $\{1, \ldots, n\}$ or a mini-batch of random data.

Below, we present the convergence analysis for smooth and non-smooth, convex and non-convex objective functions.

3.1.1 Smooth Convex Functions

For a point $\mathbf{w}$, convergence is typically measured by the objective gap:

$$g(\mathbf{w}) - \min_{\mathbf{w}} g(\mathbf{w}) = g(\mathbf{w}) - g(\mathbf{w}_*),$$

where $\mathbf{w}_*$ denotes a global optimal solution. A convergence analysis aims to show that after T iterations of updates, we can obtain a solution $\hat{\mathbf{w}}_T$ such that the expected objective gap is bounded by

$$\mathbb{E}\left[g(\hat{\mathbf{w}}_T) - g(\mathbf{w}_*)\right] \leq O\left(\frac{1}{T^\alpha}\right), \tag{3.5}$$

for some $\alpha > 0$. The term $1/T^\alpha$ is referred to as the *convergence rate*. Accordingly, to guarantee a small objective gap $\mathbb{E}[g(\hat{\mathbf{w}}_T) - g(\mathbf{w}_*)] \leq \epsilon$ for some $\epsilon \ll 1$, the bound implies that $T = O\left(\frac{1}{\epsilon^{1/\alpha}}\right)$, which is known as the iteration complexity.

Let us first assume that g is smooth and its stochastic gradient $\nabla g(\mathbf{w}; \zeta)$ satisfies the following assumption.

Assumption 3.2. *(i) $g(\mathbf{w})$ is L-smooth and convex; (ii) For any $\mathbf{w}$, we have*

$$\mathbb{E}[\|\nabla g(\mathbf{w}; \zeta) - \nabla g(\mathbf{w})\|_2^2] \leq \sigma^2$$

for some $\sigma \geq 0$.

The following lemma is useful for convergence analysis.

Lemma 3.1 *Consider the update (3.2). For any $\mathbf{w}$ we have*

$$\nabla g(\mathbf{w}_t; \zeta_t)^\top (\mathbf{w}_{t+1} - \mathbf{w}) \leq \frac{1}{2\eta_t}\|\mathbf{w} - \mathbf{w}_t\|_2^2 - \frac{1}{2\eta_t}\|\mathbf{w} - \mathbf{w}_{t+1}\|_2^2 - \frac{1}{2\eta_t}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2.$$

Proof. Since the problem (3.4) is $1/\eta_t$ strongly convex and has an optimal solution $\mathbf{w}_{t+1}$, following (1.18) for any $\mathbf{w}$ we have

$$\begin{aligned}
&\nabla g(\mathbf{w}_t; \zeta_t)^\top (\mathbf{w} - \mathbf{w}_t) + \frac{1}{2\eta_t}\|\mathbf{w} - \mathbf{w}_t\|_2^2 \\
&\geq \nabla g(\mathbf{w}_t; \zeta_t)^\top (\mathbf{w}_{t+1} - \mathbf{w}_t) + \frac{1}{2\eta_t}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 + \frac{1}{2\eta_t}\|\mathbf{w} - \mathbf{w}_{t+1}\|_2^2.
\end{aligned}$$

Re-arranging the inequality, we have

$$\nabla g(\mathbf{w}_t; \zeta_t)^\top (\mathbf{w}_{t+1} - \mathbf{w}) \leq \frac{1}{2\eta_t}\|\mathbf{w} - \mathbf{w}_t\|_2^2 - \frac{1}{2\eta_t}\|\mathbf{w} - \mathbf{w}_{t+1}\|_2^2 - \frac{1}{2\eta_t}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2.$$

□

The following lemma shows one-step objective gap bound.

Lemma 3.2 *Suppose Assumption 3.1 and 3.2 hold. For one step SGD update* $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \nabla g(\mathbf{w}_t; \xi_t)$*, we have*

$$\mathbb{E}[g(\mathbf{w}_{t+1}) - g(\mathbf{w}_*)] \leq \mathbb{E}\left[\frac{1}{2\eta_t}\|\mathbf{w}_* - \mathbf{w}_t\|_2^2 - \frac{1}{2\eta_t}\|\mathbf{w}_* - \mathbf{w}_{t+1}\|_2^2\right] + \eta_t \sigma^2.$$

Proof. From Lemma 3.1, we have

$$\begin{aligned}\nabla g(\mathbf{w}_t)^\top(\mathbf{w}_{t+1} - \mathbf{w}) \leq &\frac{1}{2\eta_t}\|\mathbf{w} - \mathbf{w}_t\|_2^2 - \frac{1}{2\eta_t}\|\mathbf{w} - \mathbf{w}_{t+1}\|_2^2 - \frac{1}{2\eta_t}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 \\ &+ (\nabla g(\mathbf{w}_t) - \nabla g(\mathbf{w}_t; \zeta_t))^\top(\mathbf{w}_{t+1} - \mathbf{w}).\end{aligned} \tag{3.6}$$

By the smoothness and convexity of g, we have

$$\begin{aligned}g(\mathbf{w}_{t+1}) &\leq g(\mathbf{w}_t) + \nabla g(\mathbf{w}_t)^\top(\mathbf{w}_{t+1} - \mathbf{w}_t) + \frac{L}{2}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 \\ &\leq g(\mathbf{w}) + \nabla g(\mathbf{w}_t)^\top(\mathbf{w}_t - \mathbf{w}) + \nabla g(\mathbf{w}_t)^\top(\mathbf{w}_{t+1} - \mathbf{w}_t) + \frac{L}{2}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 \\ &\leq g(\mathbf{w}) + \nabla g(\mathbf{w}_t)^\top(\mathbf{w}_{t+1} - \mathbf{w}) + \frac{L}{2}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2.\end{aligned} \tag{3.7}$$

Combining this with (3.6), we have

$$\begin{aligned}g(\mathbf{w}_{t+1}) - g(\mathbf{w}) \leq &\frac{1}{2\eta_t}\|\mathbf{w} - \mathbf{w}_t\|_2^2 - \frac{1}{2\eta_t}\|\mathbf{w} - \mathbf{w}_{t+1}\|_2^2 - \frac{1}{2\eta_t}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 \\ &+ \frac{L}{2}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 + (\nabla g(\mathbf{w}_t) - \nabla g(\mathbf{w}_t; \zeta_t))^\top(\mathbf{w}_{t+1} - \mathbf{w}).\end{aligned} \tag{3.8}$$

Then if $\eta_t \leq 1/L$ and plugging $\mathbf{w} = \mathbf{w}_*$, we have

$$\begin{aligned}g(\mathbf{w}_{t+1}) - g(\mathbf{w}_*) \leq &\frac{1}{2\eta_t}\|\mathbf{w}_* - \mathbf{w}_t\|_2^2 - \frac{1}{2\eta_t}\|\mathbf{w}_* - \mathbf{w}_{t+1}\|_2^2 \\ &+ (\nabla g(\mathbf{w}_t) - \nabla g(\mathbf{w}_t; \zeta_t))^\top(\mathbf{w}_{t+1} - \mathbf{w}_*).\end{aligned}$$

The challenge lies at handling the last term where $\mathbf{w}_{t+1}$ depends on ζ_t, hence its expectation is not equal to zero. To bound the last term, we introduce

$$\hat{\mathbf{w}}_{t+1} = \arg\min_{\mathbf{w}} \nabla g(\mathbf{w}_t)^\top(\mathbf{w} - \mathbf{w}_t) + \frac{1}{2\eta_t}\|\mathbf{w} - \mathbf{w}_t\|_2^2.$$

Note that $\hat{\mathbf{w}}_{t+1}$ is independent of ζ_t. Then $\mathbb{E}_{\zeta_t}[(\nabla g(\mathbf{w}_t) - \nabla g(\mathbf{w}_t; \zeta_t))^\top(\hat{\mathbf{w}}_{t+1} - \mathbf{w}_*)] = 0$. Thus, we have

$$\begin{aligned}\mathbb{E}[g(\mathbf{w}_{t+1}) - g(\mathbf{w}_*)] \leq &\mathbb{E}\left[\frac{1}{2\eta_t}\|\mathbf{w}_* - \mathbf{w}_t\|_2^2 - \frac{1}{2\eta_t}\|\mathbf{w}_* - \mathbf{w}_{t+1}\|_2^2\right] \\ &+ \mathbb{E}[(\nabla g(\mathbf{w}_t) - \nabla g(\mathbf{w}_t; \zeta_t))^\top(\mathbf{w}_{t+1} - \hat{\mathbf{w}}_{t+1})].\end{aligned}$$

Due to Lemma 1.7, we have $\|\mathbf{w}_{t+1} - \hat{\mathbf{w}}_{t+1}\|_2 \leq \eta_t \|\nabla g(\mathbf{w}_t) - \nabla g(\mathbf{w}_t; \zeta_t)\|_2$, thus

$$\mathbb{E}[g(\mathbf{w}_{t+1}) - g(\mathbf{w}_*)] \leq \mathbb{E}\left[\frac{1}{2\eta_t}\|\mathbf{w}_* - \mathbf{w}_t\|_2^2 - \frac{1}{2\eta_t}\|\mathbf{w}_* - \mathbf{w}_{t+1}\|_2^2\right] + \eta_t \sigma^2.$$

□

Theorem 3.1 *Suppose Assumption 3.1 and 3.2 hold. Let the learning rate* $\{\eta_t\}$ *be* $\eta_t = \eta \leq 1/L$ *and* $\bar{\mathbf{w}}_T = \frac{1}{T}\sum_{t=1}^{T} \mathbf{w}_{t+1}$. *Then after* T *iterations of SGD update we have*

$$\mathbb{E}\left[g(\bar{\mathbf{w}}_T) - g(\mathbf{w}_*)\right] \leq \frac{\|\mathbf{w}_1 - \mathbf{w}_*\|_2^2}{2\eta T} + \eta\sigma^2. \tag{3.9}$$

If $\eta = \min(\frac{1}{L}, \frac{\|\mathbf{w}_1 - \mathbf{w}_*\|_2}{\sqrt{2T}\sigma})$, *then*

$$\mathbb{E}\left[g(\bar{\mathbf{w}}_T) - g(\mathbf{w}_*)\right] \leq \frac{\sqrt{2}\sigma\|\mathbf{w}_1 - \mathbf{w}_*\|_2}{\sqrt{T}} + \frac{L\|\mathbf{w}_1 - \mathbf{w}_*\|_2^2}{T}.$$

Why it matters

In the convergence upper bound (3.9), the first term captures the optimization error due to the finite time horizon, while the second term represents the error induced by stochastic gradient noise.

If $\sigma = 0$ (no noise), SGD reduces to gradient descent, then a constant step size $\eta = 1/L$ can be used and the convergence rate becomes $O\left(\frac{L\|\mathbf{w}_1 - \mathbf{w}_*\|_2^2}{T}\right)$. If $\sigma^2 > 0$ (there is noise in stochastic gradient), in order to guarantee convergence, we have to set $\eta_t \to 0$ or a small value to guarantee certain level of accuracy.

For a fixed number of iterations T, a smaller variance σ allows for faster convergence with a larger learning rate η (up to a certain limit).

The iteration complexity required to achieve $\mathbb{E}[g(\bar{\mathbf{w}}_T) - g(\mathbf{w}_*)] \leq \epsilon$ is

$$T = O\left(\max\left(\frac{\sigma^2\|\mathbf{w}_1 - \mathbf{w}_*\|_2^2}{\epsilon^2}, \frac{L\|\mathbf{w}_1 - \mathbf{w}_*\|_2^2}{\epsilon}\right)\right).$$

If a mini-batch of size B is used to compute the stochastic gradient at each iteration, the variance of the stochastic gradient decreases by a factor of B. This implies that increasing the batch size, up to a certain point, can reduce the number of iterations needed.

Finally, the result also highlights that the initial learning rate η cannot be too large; in practice, an excessively large initial learning rate may cause the algorithm to diverge.

Proof. If $\eta_t = \eta$, summing the inequalities in Lemma 3.2 over $t = 1, \ldots, T$, we have

$$\mathbb{E}\Big[\sum_{t=1}^{T}(g(\mathbf{w}_{t+1})-g(\mathbf{w}_*))\Big] \leq \mathbb{E}\left[\sum_{t=1}^{T}\frac{1}{2\eta}\|\mathbf{w}_*-\mathbf{w}_t\|_2^2-\frac{1}{2\eta}\|\mathbf{w}_*-\mathbf{w}_{t+1}\|_2^2\right] + T\eta\sigma^2.$$

The first term in $[\cdot]$ is a telescoping series,

$$\sum_{t=1}^{T}\frac{1}{2\eta}\|\mathbf{w}_*-\mathbf{w}_t\|_2^2-\frac{1}{2\eta}\|\mathbf{w}_*-\mathbf{w}_{t+1}\|_2^2 \leq \frac{1}{2\eta}\|\mathbf{w}_*-\mathbf{w}_1\|_2^2-\frac{1}{2\eta}\|\mathbf{w}_*-\mathbf{w}_{T+1}\|_2^2 \leq \frac{1}{2\eta}\|\mathbf{w}_*-\mathbf{w}_1\|_2^2.$$

As a result,

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}(g(\mathbf{w}_{t+1})-g(\mathbf{w}_*))\right] \leq \frac{1}{2\eta T}\|\mathbf{w}_*-\mathbf{w}_1\|_2^2+\eta\sigma^2,$$

which concludes the proof of the first part of the theorem.

For the second part, optimizing the upper bound over η gives $\eta_* = \frac{\|\mathbf{w}_1-\mathbf{w}_*\|}{\sqrt{2T}\sigma}$. If $\eta_* \leq 1/L$, i.e., $T \geq \frac{\|\mathbf{w}_1-\mathbf{w}_*\|_2^2L^2}{2\sigma^2}$, we set $\eta=\eta_*$, then

$$\mathbb{E}\left[g(\bar{\mathbf{w}}_T)-g(\mathbf{w}_*)\right] \leq \frac{2\sigma\|\mathbf{w}_1-\mathbf{w}_*\|_2}{\sqrt{2T}}.$$

If $\eta_* > 1/L$, i.e., $\sigma^2 \leq \frac{\|\mathbf{w}_1-\mathbf{w}_*\|_2^2L^2}{2T}$, we set $\eta = 1/L$, then

$$\mathbb{E}\left[g(\bar{\mathbf{w}}_T)-g(\mathbf{w}_*)\right] \leq \frac{L\|\mathbf{w}_1-\mathbf{w}_*\|_2^2}{2T}+\frac{L\|\mathbf{w}_1-\mathbf{w}_*\|_2^2}{2T} = \frac{L\|\mathbf{w}_1-\mathbf{w}_*\|_2^2}{T}.$$

□

3.1.2 Non-smooth Convex Functions

Now, let us consider the SGD update (3.3) for non-smooth convex functions under the following assumption.

Assumption 3.3. *(i) $g(\mathbf{w})$ is convex; (ii) For any $\mathbf{w}$, we have $\mathbb{E}[\|\mathcal{G}(\mathbf{w};\zeta)\|_2^2] \leq G^2$.*

Lemma 3.3 *Suppose Assumption 3.1 and 3.3 hold. For one step SGD update $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t\mathcal{G}(\mathbf{w}_t;\xi_t)$, we have*

$$\mathbb{E}[g(\mathbf{w}_t)-g(\mathbf{w}_*)] \leq \mathbb{E}\left[\frac{1}{2\eta_t}\|\mathbf{w}_*-\mathbf{w}_t\|_2^2-\frac{1}{2\eta_t}\|\mathbf{w}-\mathbf{w}_{t+1}\|_2^2\right]+\frac{\eta_t}{2}G^2.$$

Proof. From Lemma 3.1, we have

$$\begin{aligned}\mathcal{G}(\mathbf{w}_t;\zeta_t)^\top(\mathbf{w}_t-\mathbf{w}_*) &\le \frac{1}{2\eta_t}\|\mathbf{w}_*-\mathbf{w}_t\|_2^2-\frac{1}{2\eta_t}\|\mathbf{w}_*-\mathbf{w}_{t+1}\|_2^2-\frac{1}{2\eta_t}\|\mathbf{w}_{t+1}-\mathbf{w}_t\|_2^2\\ &\quad+\mathcal{G}(\mathbf{w}_t;\zeta_t)^\top(\mathbf{w}_{t+1}-\mathbf{w}_t)\\ &\le \frac{1}{2\eta_t}\|\mathbf{w}_*-\mathbf{w}_t\|_2^2-\frac{1}{2\eta_t}\|\mathbf{w}_*-\mathbf{w}_{t+1}\|_2^2-\frac{1}{2\eta_t}\|\mathbf{w}_{t+1}-\mathbf{w}_t\|_2^2\\ &\quad+\frac{\eta_t}{2}\|\mathcal{G}(\mathbf{w}_t;\zeta_t)\|_2^2+\frac{1}{2\eta_t}\|\mathbf{w}_{t+1}-\mathbf{w}_t\|_2^2,\end{aligned} \tag{3.10}$$

where the last inequality uses the Young's inequality. Taking expectation on both sides, we have

$$\mathbb{E}[\mathcal{G}(\mathbf{w}_t;\zeta_t)^\top(\mathbf{w}_t-\mathbf{w}_*)] \le \mathbb{E}\left[\frac{1}{2\eta_t}\|\mathbf{w}_*-\mathbf{w}_t\|_2^2-\frac{1}{2\eta_t}\|\mathbf{w}-\mathbf{w}_{t+1}\|_2^2\right]+\frac{\eta_t}{2}G^2. \tag{3.11}$$

Since $\mathbf{w}_t$ is independent of ζ_t, we have $\mathbb{E}[\mathcal{G}(\mathbf{w}_t;\zeta_t)^\top(\mathbf{w}_t-\mathbf{w}_*)] = \mathbb{E}[\mathbf{v}_t^\top(\mathbf{w}_t-\mathbf{w}_*)]$ for some $\mathbf{v}_t \in \partial g(\mathbf{w}_t)$. By the convexity of g, we have

$$\begin{aligned}&\mathbb{E}[g(\mathbf{w}_t)-g(\mathbf{w}_*)] \le \mathbb{E}[\mathbf{v}_t^\top(\mathbf{w}_t-\mathbf{w}_*)] = \mathbb{E}[\mathcal{G}(\mathbf{w}_t;\zeta_t)^\top(\mathbf{w}_t-\mathbf{w}_*)]\\ &\le \mathbb{E}\left[\frac{1}{2\eta_t}\|\mathbf{w}_*-\mathbf{w}_t\|_2^2-\frac{1}{2\eta_t}\|\mathbf{w}-\mathbf{w}_{t+1}\|_2^2\right]+\frac{\eta_t}{2}G^2.\end{aligned} \tag{3.12}$$

□

The theorem below establishes the convergence of SGD for non-smooth convex functions as measured by the objective gap.

Theorem 3.2 *Suppose Assumption 3.1 and 3.3 hold. Let the learning rate $\{\eta_t\}$ be $\eta_t = \eta$ and $\bar{\mathbf{w}}_T = \frac{1}{T}\sum_{t=1}^T \mathbf{w}_t$. Then after for T iterations of SGD update (3.3) we have*

$$\mathbb{E}\left[g(\bar{\mathbf{w}}_T)-g(\mathbf{w}_*)\right] \le \frac{\|\mathbf{w}_1-\mathbf{w}_*\|_2^2}{2\eta T}+\frac{\eta G^2}{2}.$$

If $\eta = \frac{\|\mathbf{w}_1-\mathbf{w}_*\|_2}{\sqrt{T}G}$, then

$$\mathbb{E}\left[g(\bar{\mathbf{w}}_T)-g(\mathbf{w}_*)\right] \le \frac{G\|\mathbf{w}_1-\mathbf{w}_*\|_2}{\sqrt{T}}.$$

Why it matters

The above theorem exhibits the key difference in the convergence of SGD for smooth convex functions and non-smooth convex functions. Even with a zero variance for the stochastic subgradient , the convergence rate is still $O(1/\sqrt{T})$. The reason is that for smooth convex functions when $g(\mathbf{w}) \to g(\mathbf{w}_*)$, we have

$\nabla g(\mathbf{w}) \to 0$ (cf. Lemma 1.5(b)), which is not true for non-smooth convex functions.

Proof. The proof is similar to that in the smooth case. □

3.1.3 Smooth Non-Convex Functions

For a non-convex function, it is generally NP-hard to find a global optimal solution. Hence, our goal here is to establish the complexity of SGD for finding an ϵ-stationary solution with $\epsilon \ll 1$, as defined below.

Definition 3.1 (ϵ-stationary solution) $\mathbf{w}$ is an ϵ-stationary solution to $\min_{\mathbf{w}} g(\mathbf{w})$, if $\|\nabla g(\mathbf{w})\|_2 \leq \epsilon$.

Assumption 3.4. *(i) $g(\mathbf{w})$ is L-smooth and non-convex; (ii) For any $\mathbf{w}$, we have*

$$\mathbb{E}[\|\nabla g(\mathbf{w}; \zeta) - \nabla g(\mathbf{w})\|_2^2] \leq \sigma^2$$

for some $\sigma \geq 0$.

Based on the above assumptions, we establish the following convergence guarantee.

Theorem 3.3 *Suppose Assumption 3.1 and 3.4 hold. Let the learning rate $\{\eta_t\}$ be $\eta_t = \min\{\frac{1}{L}, \frac{D}{\sigma\sqrt{T}}\}$ for some constant $D > 0$. Let $\tau \in \{1, \ldots, T\}$ be a random sample following a distribution $\Pr(\tau = t) = \frac{1}{T}$. Then we have*

$$\mathbb{E}[\|\nabla g(\mathbf{w}_\tau)\|_2^2] \leq \frac{2L(g(\mathbf{w}_1) - g(\mathbf{w}_*)}{T} + \left(\frac{2(g(\mathbf{w}_1) - g(\mathbf{w}_*))}{D} + DL\right)\frac{\sigma}{\sqrt{T}}.$$

Proof. For brevity of notation, we let $\nabla g_t(\mathbf{w}_t) = \nabla g(\mathbf{w}_t; \zeta_t)$. Due to the L-smoothness of g, we have

$$
\begin{aligned}
&g(\mathbf{w}_{t+1}) \leq g(\mathbf{w}_t) + \nabla g(\mathbf{w}_t)^\top (\mathbf{w}_{t+1} - \mathbf{w}_t) + \frac{L}{2} \|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 \\
&= g(\mathbf{w}_t) - \eta_t \nabla g(\mathbf{w}_t)^\top \nabla g_t(\mathbf{w}_t) + \frac{\eta_t^2 L}{2} \|\nabla g_t(\mathbf{w}_t)\|_2^2 \\
&= g(\mathbf{w}_t) - \eta_t \|\nabla g(\mathbf{w}_t)\|_2^2 + \eta_t \nabla g(\mathbf{w}_t)^\top (\nabla g(\mathbf{w}_t) - \nabla g_t(\mathbf{w}_t)) + \frac{\eta_t^2 L}{2} \|\nabla g_t(\mathbf{w}_t)\|_2^2 \\
&= g(\mathbf{w}_t) - \eta_t \|\nabla g(\mathbf{w}_t)\|_2^2 + \eta_t \nabla g(\mathbf{w}_t)^\top (\nabla g(\mathbf{w}_t) - \nabla g_t(\mathbf{w}_t)) \\
&+ \frac{\eta_t^2 L}{2} \|\nabla g_t(\mathbf{w}_t) - \nabla g(\mathbf{w}_t) + \nabla g(\mathbf{w}_t)\|_2^2 \\
&= g(\mathbf{w}_t) - (\eta_t - \frac{\eta_t^2 L}{2}) \|\nabla g(\mathbf{w}_t)\|_2^2 + (\eta_t - \eta_t^2 L) \nabla g(\mathbf{w}_t)^\top (\nabla g(\mathbf{w}_t) - \nabla g_t(\mathbf{w}_t)) \\
&+ \frac{\eta_t^2 L}{2} \|\nabla g_t(\mathbf{w}_t) - \nabla g(\mathbf{w}_t)\|_2^2 .
\end{aligned}
$$

Taking expectation over ζ_t given $\mathbf{w}_t$ on both sides, we have

$$
\mathbb{E}_{\zeta_t} \left[g(\mathbf{w}_{t+1}) \right] \leq g(\mathbf{w}_t) - (\eta_t - \frac{\eta_t^2 L}{2}) \|\nabla g(\mathbf{w}_t)\|_2^2 + \frac{\eta_t^2 L}{2} \sigma^2. \tag{3.13}
$$

Telescoping this from $t = 1$ to T gives

$$
\mathbb{E} \left[\sum_{t=1}^{T} (\eta_t - \frac{\eta_t^2 L}{2}) \|\nabla g(\mathbf{w}_t)\|_2^2 \right] \leq (g(\mathbf{w}_1) - g(\mathbf{w}_*) + \sum_{t=1}^{T} \frac{\eta_t^2 L}{2} \sigma^2.
$$

As a result,

$$
\mathbb{E} \left[\|\nabla g(\mathbf{w}_\tau)\|_2^2 \right] \leq \frac{(g(\mathbf{w}_1) - g(\mathbf{w}_*))}{\sum_{t=1}^{T} (\eta_t - \frac{\eta_t^2 L}{2})} + \frac{\sum_{t=1}^{T} \eta_t^2 L}{2 \sum_{t=1}^{T} (\eta_t - \frac{\eta_t^2 L}{2})} \sigma^2.
$$

Plugging the value of $\eta_t = \min(\frac{1}{L}, \frac{D}{\sigma\sqrt{T}})$, we have

$$
\begin{aligned}
\mathbb{E} \left[\|\nabla g(\mathbf{w}_\tau)\|_2^2 \right] &\leq \frac{g(\mathbf{w}_1) - g(\mathbf{w}_*)}{T(\eta_1 - \frac{\eta_1^2 L}{2})} + \frac{T \eta_1^2 L}{2T(\eta_1 - \frac{\eta_1^2 L}{2})} \sigma^2 \\
&\leq \frac{2(g(\mathbf{w}_1) - g(\mathbf{w}_*))}{T\eta_1} + \eta_1 L \sigma^2 \\
&\leq \max \left(\frac{2L(g(\mathbf{w}_1) - g(\mathbf{w}_*))}{T}, \frac{2(g(\mathbf{w}_1) - g(\mathbf{w}_*))\sigma}{D\sqrt{T}} \right) + \frac{D\sigma L}{\sqrt{T}} \\
&\leq \frac{2L(g(\mathbf{w}_1) - g(\mathbf{w}_*))}{T} + \left(\frac{2(g(\mathbf{w}_1) - g(\mathbf{w}_*))}{D} + DL \right) \frac{\sigma}{\sqrt{T}}.
\end{aligned}
$$

If we set $\eta_t = \min(\frac{1}{L}, \frac{D}{\sigma\sqrt{t}})$, then $\sum_{t=1}^{T} \eta_t \geq \Omega(\sqrt{T})$ and $\sum_{t=1}^{T} \eta_t^2 \leq O(\log(T))$, then $\mathbb{E} \left[\|\nabla g(\mathbf{w}_\tau)\|_2^2 \right] \leq O(\log T / T)$. □

3.1.4 Non-smooth Weakly Convex Functions

Next, let us extend the analysis to non-smooth non-convex functions. Consider a function $g : \mathbb{R}^d \mapsto \mathbb{R}$ and a point $\mathbf{w} \in \mathbb{R}^d$ with $g(\mathbf{w})$ finite. The Fréchet subdifferential of g at $\mathbf{w}$, denoted $\partial g(\mathbf{w})$, consists of all vectors $\mathbf{v}$ satisfying

$$g(\mathbf{w}) \geq g(\mathbf{w}') + \mathbf{v}^\top(\mathbf{w} - \mathbf{w}') + o(\|\mathbf{w} - \mathbf{w}'\|_2) \text{ as } \mathbf{w}' \to \mathbf{w}.$$

We consider a family of non-convex functions, namely weakly convex functions. A lower semi-continuous function g is called ρ-weakly, if there exists $\rho > 0$ such that:

$$g(\mathbf{w}) \geq g(\mathbf{w}') + \mathbf{v}^\top(\mathbf{w} - \mathbf{w}') - \frac{\rho}{2}\|\mathbf{w} - \mathbf{w}'\|_2^2, \quad \forall \mathbf{w}, \mathbf{w}', \mathbf{v} \in \partial g(\mathbf{w}').$$

It is easy to show that if g is ρ-weakly convex, then $g(\mathbf{w}) + \frac{\rho}{2}\|\mathbf{w}\|_2^2$ is a convex function of $\mathbf{w}$. A smooth function is weakly convex, but the reverse is not necessarily true.

Example

Example 3.1 (Compositional functions). *Let $F(\mathbf{x}) = f(g(\mathbf{x}))$. If f convex and G_1-Lipschitz continuous and $g(\mathbf{x})$ is L_2-smooth, then F is ρ-weakly convex for some $\rho > 0$. We will prove this in Section 5.3. The OCE risk (2.22) is a special case when ϕ^* is non-smooth and the loss function $\ell(\mathbf{w}; \mathbf{z})$ is smooth non-convex.*

Example 3.2 (Compositional functions). *Let $F(\mathbf{x}) = f(g(\mathbf{x}))$. If f L_1-smooth and monotonically non-decreasing and $g(\mathbf{x})$ is non-smooth convex and G_2-Lipschitz continuous, then F is ρ-weakly convex for some $\rho > 0$.*
Let us prove it. Since $f(g)$ is L_1 smooth, i.e., for any $\mathbf{w}, \mathbf{v} \in \mathbb{R}^d$, we have $f(g(\mathbf{v})) + f'(g(\mathbf{v}))(g(\mathbf{w}) - g(\mathbf{v})) - \frac{L_1}{2}|g(\mathbf{w}) - g(\mathbf{v})|^2 \leq f(g(\mathbf{w}))$. Since g is convex, i.e. for any $\mathbf{w}, \mathbf{v} \in \mathbb{R}^d$, $g(\mathbf{w}) \geq g(\mathbf{v}) + \partial g(\mathbf{v})^\top(\mathbf{w} - \mathbf{v})$, then

$$\begin{aligned} f(g(\mathbf{w})) - f(g(\mathbf{v})) \geq & f'(g(\mathbf{v}))\partial g(\mathbf{v})^\top(\mathbf{w} - \mathbf{v}) - \frac{L_1}{2}|g(\mathbf{w}) - g(\mathbf{v})|^2 \\ \geq & f'(g(\mathbf{v}))\partial g(\mathbf{v})^\top(\mathbf{w} - \mathbf{v}) - \frac{G_2^2 L_1}{2}\|\mathbf{w} - \mathbf{v}\|_2^2, \end{aligned}$$

where the first inequality uses $f'(g(\mathbf{v})) \geq 0$; the second inequality uses the fact that $\|\partial g(\mathbf{w})\|_2 \leq G_2$. That is, $f(g(\mathbf{w}))$ is G^2L-weakly convex.
An important application of this function in machine learning is optimizing the truncation of a convex loss $g(\mathbf{w}) = \ell(\mathbf{w}; \mathbf{z}) \geq 0$ with a smooth truncation function $f(\ell(\mathbf{w}; \mathbf{z})) = \alpha \log(1 + \frac{\ell(\mathbf{w};\mathbf{z})}{\alpha})$ for some $\alpha > 0$, which is useful for tackling heavy-tailed data distribution.

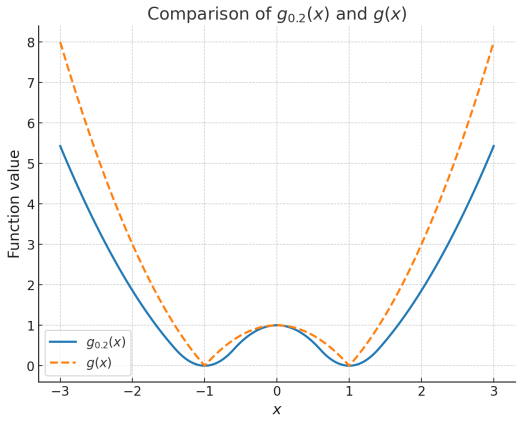


Fig. 3.1: Moreau envelope of $g(x) = |x^2 - 1|$ with $\lambda = 0.2$.

Nearly ϵ-stationary solution

When $g(\cdot)$ is non-smooth, finding an ϵ-stationary solution such that $\|\nabla g(\mathbf{w})\|_2 \leq \epsilon$ is difficult even for a convex function. Let us consider a simple example $\min_w |w|$. The only stationary point is the optimal solution $w_* = 0$, and any $w \neq 0$ is not an ϵ-stationary solution ($\epsilon < 1$) no matter how close w to 0. To address this issue, we introduce a weak notion of ϵ-stationary solution, termed nearly ϵ-stationary solution.

Definition 3.2 (Nearly ϵ-stationary solution) $\mathbf{w}$ is a nearly ϵ-stationary solution to $\min_{\mathbf{w}} g(\mathbf{w})$, if there exists $\hat{\mathbf{w}}$ such that $\|\mathbf{w} - \hat{\mathbf{w}}\| \leq O(\epsilon)$ and $\text{dist}(0, \partial g(\hat{\mathbf{w}})) \leq \epsilon$.

A useful tool for deriving a nearly ϵ-stationary solution is the Moreau envelope of g:

$$g_\lambda(\mathbf{w}) := \min_{\mathbf{u}} g(\mathbf{u}) + \frac{1}{2\lambda}\|\mathbf{u} - \mathbf{w}\|_2^2. \tag{3.14}$$

Define

$$\text{prox}_{\lambda g}(\mathbf{w}) := \arg\min_{\mathbf{u}} g(\mathbf{u}) + \frac{1}{2\lambda}\|\mathbf{u} - \mathbf{w}\|_2^2. \tag{3.15}$$

An example of a weakly convex function and its Moreau envelope is illustrated in Figure 3.1.

The proposition below shows that when λ is sufficiently small, $g_\lambda(\cdot)$ is a smooth function.

Proposition 3.1. *Consider a ρ-weakly convex function $g(\cdot)$. Then for any $\lambda \in (0, \rho^{-1})$, the Moreau envelope $g_\lambda(\cdot)$ is $\frac{2-\lambda\rho}{\lambda(1-\lambda\rho)}$-smooth, with gradient given by*

$$\nabla g_\lambda(\mathbf{w}) = \frac{1}{\lambda}(\mathbf{w} - prox_{\lambda g}(\mathbf{w})).$$

Proof. First, when $\lambda < \rho^{-1}$ we have $g(\mathbf{u}) + \frac{1}{2\lambda}\|\mathbf{u} - \mathbf{w}\|_2^2$ become $(\frac{1}{\lambda} - \rho)$ -strongly convex. Hence the solution $\text{prox}_{\lambda g}(\mathbf{w})$ is unique for a given $\mathbf{w}$. We can also write $\text{prox}_{\lambda g}(\mathbf{w})$ as

$$\begin{aligned}\text{prox}_{\lambda g}(\mathbf{w}) &:= \arg\min_{\mathbf{u}} g(\mathbf{u}) + \frac{1}{2\lambda}\|\mathbf{u} - \mathbf{w}\|_2^2 \\ &= \arg\min_{\mathbf{u}} \underbrace{g(\mathbf{u}) + \frac{\rho}{2}\|\mathbf{u}\|_2^2}_{r(\mathbf{u})} + \frac{1}{2}\left(\frac{1}{\lambda} - \rho\right)\left\|\mathbf{u} - \frac{1}{1-\lambda\rho}\mathbf{w}\right\|_2^2.\end{aligned}$$

Due to Lemma 1.7, we have $\|\text{prox}_{\lambda g}(\mathbf{w}) - \text{prox}_{\lambda g}(\mathbf{w}')\|_2 \leq \frac{1}{1-\lambda\rho}\|\mathbf{w} - \mathbf{w}'\|_2$. Then

$$\begin{aligned}&\|\nabla g_\lambda(\mathbf{w}) - \nabla g_\lambda(\mathbf{w}')\|_2 = \frac{1}{\lambda}\|(\mathbf{w} - \text{prox}_{\lambda g}(\mathbf{w})) - (\mathbf{w}' - \text{prox}_{\lambda g}(\mathbf{w}'))\|_2 \\ &\leq \frac{1}{\lambda}\left(\|\mathbf{w} - \mathbf{w}'\|_2 + \frac{1}{1-\lambda\rho}\|\mathbf{w} - \mathbf{w}'\|_2\right) = \frac{2-\lambda\rho}{\lambda(1-\lambda\rho)}\|\mathbf{w} - \mathbf{w}'\|_2.\end{aligned}$$

□

With the Moreau envelope, we can use the norm of its gradient to measure the convergence for optimizing the original function.

Proposition 3.2. *If $\lambda < \rho^{-1}$, we have*

$$g_\lambda(\mathbf{w}) \leq g(\mathbf{w}), \quad \min_{\mathbf{w}} g_\lambda(\mathbf{w}) = \min_{\mathbf{w}} g(\mathbf{w}). \tag{3.16}$$

If $\|\nabla g_\lambda(\mathbf{w})\|_2 \leq \epsilon$, then $\hat{\mathbf{w}} = prox_{\lambda g}(\mathbf{w})$ is a nearly ϵ-stationary solution. In particular,

$$\begin{aligned}&\|\hat{\mathbf{w}} - \mathbf{w}\|_2 = \lambda\|\nabla g_\lambda(\mathbf{w})\|_2 \leq \lambda\epsilon, \\ &dist(0, \partial g(\hat{\mathbf{w}})) \leq \|\nabla g_\lambda(\mathbf{w})\|_2 \leq \epsilon.\end{aligned} \tag{3.17}$$

Proof. $g_\lambda(\mathbf{w}) \leq g(\mathbf{w})$ follows the definition of $g_\lambda(\mathbf{w})$. Then $g_\lambda(\mathbf{w}_*) \leq g(\mathbf{w}_*)$. To prove they are equal, we have

$$g_\lambda(\mathbf{w}) = \min_{\mathbf{u}} g(\mathbf{u}) + \frac{1}{2\lambda}\|\mathbf{u} - \mathbf{w}\|_2^2 \geq \min_{\mathbf{u}} g(\mathbf{u}) = g(\mathbf{w}_*).$$

Since $\nabla g_\lambda(\mathbf{w}) = \frac{1}{\lambda}(\mathbf{w} - \hat{\mathbf{w}})$, which implies the second inequality. The first inequality is due to the first-order optimality condition of $\min_{\mathbf{u}} g(\mathbf{u}) + \frac{1}{2\lambda}\|\mathbf{u} - \mathbf{w}\|_2^2$. □

Why it matters

Proposition 3.2 shows that if we can make $\|\nabla g_\lambda(\mathbf{w})\|_2$ small, then $\mathbf{w}$ is close to an ϵ-stationary solution $\hat{\mathbf{w}}$ of the original function $g(\mathbf{w})$. The smaller the λ, the closer between $\mathbf{w}$ and $\hat{\mathbf{w}}$.

Convergence Analysis

Assumption 3.5. *(i) $g(\mathbf{w})$ is ρ-weakly convex; (ii) For any $\mathbf{w}$, $\mathbb{E}_\zeta[\|\mathcal{G}(\mathbf{w},\zeta)\|_2^2] \le G^2$ for some $G \ge 0$.*

Lemma 3.4 *Let us consider an update $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{z}_t$. If $\mathbb{E}_t[\mathbf{z}_t] = \mathcal{M}_t$ and $\mathbb{E}_t[\|\mathbf{z}_t\|_2^2] \le G^2$, then we have*

$$\mathbb{E}_t[g_\lambda(\mathbf{w}_{t+1})] \le g_\lambda(\mathbf{w}_t) + \frac{\eta_t}{\lambda}(\hat{\mathbf{w}}_t - \mathbf{w}_t)^\top \mathcal{M}_t + \frac{\eta_t^2 G^2}{2\lambda},$$

where $\hat{\mathbf{w}}_t = prox_{\lambda g}(\mathbf{w}_t)$.

Proof. We have

$$\begin{aligned} g_\lambda(\mathbf{w}_{t+1}) &= g(\hat{\mathbf{w}}_{t+1}) + \frac{1}{2\lambda}\|\hat{\mathbf{w}}_{t+1} - \mathbf{w}_{t+1}\|_2^2 \le g(\hat{\mathbf{w}}_t) + \frac{1}{2\lambda}\|\hat{\mathbf{w}}_t - \mathbf{w}_{t+1}\|_2^2 \\ &= g(\hat{\mathbf{w}}_t) + \frac{1}{2\lambda}\|\hat{\mathbf{w}}_t - \mathbf{w}_t\|_2^2 - \frac{1}{2\lambda}\|\hat{\mathbf{w}}_t - \mathbf{w}_t\|_2^2 + \frac{1}{2\lambda}\|\hat{\mathbf{w}}_t - \mathbf{w}_{t+1}\|_2^2. \end{aligned}$$

Merging the first two terms we get $g_\lambda(\mathbf{w}_t)$, and using the three-point equality $2(a-b)(b-c) = \|a-c\|_2^2 - \|a-b\|_2^2 - \|b-c\|_2^2$ to merge the last two terms we get

$$\begin{aligned} g_\lambda(\mathbf{w}_{t+1}) &= g_\lambda(\mathbf{w}_t) + \frac{1}{\lambda}(\hat{\mathbf{w}}_t - \mathbf{w}_t)^\top(\mathbf{w}_t - \mathbf{w}_{t+1}) + \frac{1}{2\lambda}\|\mathbf{w}_t - \mathbf{w}_{t+1}\|_2^2 \\ &= g_\lambda(\mathbf{w}_t) + \frac{1}{\lambda}(\hat{\mathbf{w}}_t - \mathbf{w}_t)^\top \eta_t \mathbf{z}_t + \frac{\eta_t^2}{2\lambda}\|\mathbf{z}_t\|_2^2. \end{aligned}$$

Taking expectation over ζ_t given $\mathbf{w}_t$ on both sides, we have

$$\mathbb{E}_t[g_\lambda(\mathbf{w}_{t+1})] \le g_\lambda(\mathbf{w}_t) + \frac{1}{\lambda}(\hat{\mathbf{w}}_t - \mathbf{w}_t)^\top \eta_t \mathcal{M}_t + \frac{\eta_t^2 G^2}{2\lambda}.$$

□

Lemma 3.5 *Under the same setting of Lemma 3.4 we have*

$$\eta_t(1 - \lambda\rho)\|\nabla g_\lambda(\mathbf{w}_t)\|_2^2 \le g_\lambda(\mathbf{w}_t) - \mathbb{E}_t[g_\lambda(\mathbf{w}_{t+1})] + \frac{\eta_t^2 G^2}{2\lambda}.$$

Proof. Due to the weak convexity of g, for any $\mathcal{M}_t \in \partial g(\mathbf{w}_t)$, we have

$$\mathcal{M}_t^\top(\mathbf{w}_t - \hat{\mathbf{w}}_t) \geq g(\mathbf{w}_t) - g(\hat{\mathbf{w}}_t) - \frac{\rho}{2}\|\hat{\mathbf{w}}_t - \mathbf{w}_t\|_2^2$$
$$= (g(\mathbf{w}_t) + \frac{1}{2\lambda}\|\mathbf{w}_t - \mathbf{w}_t\|_2^2) - (g(\hat{\mathbf{w}}_t) + \frac{1}{2\lambda}\|\hat{\mathbf{w}}_t - \mathbf{w}_t\|_2^2) + (\frac{1}{2\lambda} - \frac{\rho}{2})\|\hat{\mathbf{w}}_t - \mathbf{w}_t\|_2^2.$$

Since $h(\mathbf{w}) = g(\mathbf{w}) + \frac{1}{2\lambda}\|\mathbf{w} - \mathbf{w}_t\|_2^2$ is $(1/\lambda - \rho)$-strongly convex and $\hat{\mathbf{w}}_t = \arg\min h(\mathbf{w})$, then applying Lemma 1.6(a), we get

$$(g(\mathbf{w}_t) + \frac{1}{2\lambda}\|\mathbf{w}_t - \mathbf{w}_t\|_2^2) - (g(\hat{\mathbf{w}}_t) + \frac{1}{2\lambda}\|\hat{\mathbf{w}}_t - \mathbf{w}_t\|_2^2) \geq (\frac{1}{2\lambda} - \frac{\rho}{2})\|\hat{\mathbf{w}}_t - \mathbf{w}_t\|_2^2.$$

Combining the above two inequalities we have

$$\mathcal{M}_t^\top(\mathbf{w}_t - \hat{\mathbf{w}}_t) \geq g(\mathbf{w}_t) - g(\hat{\mathbf{w}}_t) - \frac{\rho}{2}\|\hat{\mathbf{w}}_t - \mathbf{w}_t\|_2^2$$
$$\geq (\frac{1}{2\lambda} - \frac{\rho}{2})\|\hat{\mathbf{w}}_t - \mathbf{w}_t\|_2^2 + (\frac{1}{2\lambda} - \frac{\rho}{2})\|\hat{\mathbf{w}}_t - \mathbf{w}_t\|_2^2 = (\lambda - \lambda^2\rho)\|\nabla g_\lambda(\mathbf{w}_t)\|_2^2.$$

Plugging this into the inequality in Lemma 3.4, we have

$$\eta_t(1 - \lambda\rho)\|\nabla g_\lambda(\mathbf{w}_t)\|_2^2 \leq g_\lambda(\mathbf{w}_t) - \mathbb{E}_t[g_\lambda(\mathbf{w}_{t+1})] + \frac{\eta_t^2 G^2}{2\lambda}.$$

□

Theorem 3.4 *Suppose the learning rate $\{\eta_t\}$ is set to $\eta_t = \frac{C}{\sqrt{T}}$. Let $\tau \in \{1, \ldots, T\}$ be a random sample following a distribution $\Pr(\tau = t) = \frac{1}{T}$. Then for any $\lambda \in (0, \rho^{-1})$, we have*

$$\mathbb{E}[\|\nabla g_\lambda(\mathbf{w}_\tau)\|_2^2] \leq \frac{g(\mathbf{w}_1) - g(\mathbf{w}_*)}{(1 - \lambda\rho)C\sqrt{T}} + \frac{CG^2}{2\lambda(1 - \lambda\rho)\sqrt{T}}.$$

Proof. Summing up the inequalities in Lemma 3.5 over $t = 1, \ldots, T$ and taking expectation over all randomness, we have

$$\mathbb{E}\left[\sum_{t=1}^T \eta_t(1 - \lambda\rho)\|\nabla g_\lambda(\mathbf{w}_t)\|_2^2\right] \leq g(\mathbf{w}_1) - g(\mathbf{w}_*) + \sum_{t=1}^T \frac{\eta_t^2 G^2}{2\lambda}.$$

where we have used $g_\lambda(\mathbf{w}) \leq g(\mathbf{w})$ and $\min g_\lambda(\mathbf{w}) = g(\mathbf{w}_*)$. Then

$$\mathbb{E}[\|\nabla g_\lambda(\mathbf{w}_\tau)\|_2^2] \leq \frac{g(\mathbf{w}_1) - g(\mathbf{w}_*)}{(1 - \lambda\rho)C\sqrt{T}} + \frac{CG^2}{2\lambda(1 - \lambda\rho)\sqrt{T}}.$$

□

Algorithm 2 SPGD

1: **Input:** learning rate schedule $\{\eta_t\}_{t=1}^T$, starting point $\mathbf{w}_1$
2: **for** $t = 1, \ldots, T$ **do**
3: Compute an unbiased gradient estimator $\mathbf{z}_t = \nabla g(\mathbf{w}_t; \zeta_t)$
4: Update the model $\mathbf{w}$ by $\mathbf{w}_{t+1} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \mathbf{z}_t^\top(\mathbf{w} - \mathbf{w}_t) + \frac{1}{2\eta_t}\|\mathbf{w} - \mathbf{w}_t\|_2^2 + r(\mathbf{w})$.
5: **end for**

3.2 Stochastic Proximal Gradient Descent

Let us consider the following stochastic composite optimization problem:

$$\min_{\mathbf{w}\in\mathbb{R}^d} F(\mathbf{w}) := \mathbb{E}_\zeta[g(\mathbf{w};\zeta)] + r(\mathbf{w}), \tag{3.18}$$

where $g(\mathbf{w}) = \mathbb{E}_\zeta[g(\mathbf{w};\zeta)]$ is a smooth function and $r(\mathbf{w})$ is a possibly non-smooth function. In machine learning, r usually corresponds to some regularizer on the model parameter. We make the following assumption.

Assumption 3.6. *Suppose the following conditions hold:*

(i) $g(\mathbf{w})$ is L-smooth and convex, and $r(\mathbf{w})$ is convex.
(ii) There exists $\sigma > 0$ such that $\mathbb{E}_\zeta[\|\nabla g(\mathbf{w};\zeta) - \nabla g(\mathbf{w})\|_2^2] \le \sigma^2$ for all $\mathbf{w}$.

If the regularizer r is non-smooth, the overall objective function is also non-smooth. Consequently, applying SGD directly cannot exploit the smoothness of g, which would otherwise enable faster convergence and enjoy the variance scaling in the convergence bound.

To address this challenge, we can employ the stochastic proximal gradient descent (SPGD) method:

$$\begin{aligned}\mathbf{w}_{t+1} &= \arg\min_{\mathbf{w}\in\mathbb{R}^d} \nabla g(\mathbf{w}_t;\zeta_t)^\top(\mathbf{w} - \mathbf{w}_t) + r(\mathbf{w}) + \frac{1}{2\eta_t}\|\mathbf{w} - \mathbf{w}_t\|_2^2 \\ &= \arg\min_{\mathbf{w}\in\mathbb{R}^d} r(\mathbf{w}) + \frac{1}{2\eta_t}\|\mathbf{w} - (\mathbf{w}_t - \eta_t\nabla g(\mathbf{w}_t;\zeta_t))\|_2^2.\end{aligned} \tag{3.19}$$

This is also known as forward-backward splitting, where $\widetilde{\mathbf{w}}_{t+1} = \mathbf{w}_t - \eta_t\nabla g(\mathbf{w}_t;\zeta_t)$ is the forward step and the proximal mapping of r is the backward step:

$$\mathbf{w}_{t+1} = \mathrm{prox}_{\eta_t r}(\widetilde{\mathbf{w}}_{t+1}) = \arg\min_{\mathbf{w}} r(\mathbf{w}) + \frac{1}{2\eta_t}\|\mathbf{w} - \widetilde{\mathbf{w}}_{t+1}\|_2^2.$$

When r is absent, the above update is equivalent to the SGD update. If $r(\mathbf{w})$ corresponds to a domain constraint $\mathbf{w} \in \mathcal{W}$, i.e., $r(\mathbf{w}) = \mathbb{I}_{0-\infty}(\mathbf{w} \in \mathcal{W})$, the above update becomes

$$\mathbf{w}_{t+1} = \Pi_{\mathcal{W}}[\widetilde{\mathbf{w}}_{t+1}] = \min_{\mathbf{w}\in\mathcal{W}} \|\mathbf{w} - \widetilde{\mathbf{w}}_{t+1}\|_2^2, \tag{3.20}$$

Regularization	$r(\cdot)$	$\text{prox}_{\eta r}(\bar{\mathbf{w}})$ or $\text{prox}_{\eta r}(\bar{W})$
Euclidean norm square	$\frac{\lambda}{2}\|\mathbf{w}\|_2^2$	$\frac{\bar{\mathbf{w}}}{1+\lambda\eta}$
Euclidean norm	$\lambda\|\mathbf{w}\|_2$	$\left(1-\frac{\lambda\eta}{\|\bar{\mathbf{w}}\|_2}\right)_+\bar{\mathbf{w}}$
Lasso	$\lambda\|\mathbf{w}\|_1$	$\text{sign}(\bar{\mathbf{w}})\odot\max\{\|\bar{\mathbf{w}}\|-\lambda\eta,\,0\}$
Group Lasso	$\lambda\sum_g\|\mathbf{w}_g\|_2$	$\left(1-\frac{\lambda\eta}{\|\bar{\mathbf{w}}_g\|_2}\right)_+\bar{\mathbf{w}}_g$ (for each group g)
Elastic Net	$\alpha\|\mathbf{w}\|_1+\frac{\beta}{2}\|\mathbf{w}\|_2^2$	$\frac{1}{1+\eta\beta}\left(\text{sign}(\bar{\mathbf{w}})\odot\max\{\|\bar{\mathbf{w}}\|-\eta\alpha,\,0\}\right)$
Trace norm (nuclear)	$\lambda\|W\|_*=\lambda\sum_i\sigma_i(W)$	$U\,\text{diag}\left((\sigma_i-\lambda\eta)_+\right)V^\top$ $(\bar{W}=U\,\text{diag}(\sigma_i)V^\top)$

Table 3.1: Examples of regularization functions $r(\cdot)$ and their proximal mappings, where σ_i denote the i-th singular value of a matrix.

which is the projection of $\widetilde{\mathbf{w}}_{t+1} = \mathbf{w}_t - \eta_t\nabla g(\mathbf{w}_t,\zeta_t)$ onto the constrained domain $\mathcal{W}$. This is known as projected SGD method.

Explanation of SPGD update

The update (3.19) is equivalent to:

$$\mathbf{w}_{t+1} = \arg\min_{\mathbf{w}} g(\mathbf{w}_t;\zeta_t) + \nabla g(\mathbf{w}_t;\zeta_t)^\top(\mathbf{w}-\mathbf{w}_t) + r(\mathbf{w}) + \frac{1}{2\eta_t}\|\mathbf{w}-\mathbf{w}_t\|_2^2.$$

Unlike SGD, SPGD uses a stochastic linear approximation $g(\mathbf{w}_t;\zeta_t) + \nabla g(\mathbf{w}_t;\zeta_t)^\top(\mathbf{w}-\mathbf{w}_t) + r(\mathbf{w})$ as a stochastic surrogate for $g(\mathbf{w})+r(\mathbf{w})$. Using the first-order optimality condition of (3.19), $\mathbf{w}_{t+1}$ satisfies

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta(\nabla g(\mathbf{w}_t;\zeta_t) + \partial r(\mathbf{w}_{t+1})). \tag{3.21}$$

It resembles SGD but differs in that it uses a stochastic gradient of g evaluated at $\mathbf{w}_t$ and a subgradient of r evaluated at $\mathbf{w}_{t+1}$.

In order to make the update efficient, the proximal mapping of r should be easily computable. Table 3.1 presents several examples of regularizers r and the corresponding solutions of their proximal mappings, followed by explanations below. We leave the detailed derivations of these proximal mappings to the reader as exercises.

Examples

Example 3.3 (Euclidean norm square)**.** *This is the most commonly used regularizer. Its proximal mapping shrinks the magnitude of the input vector* $\bar{\mathbf{w}}$*, effectively performing weight decay.*

Example 3.4 (ℓ_1 norm)**.** *The* ℓ_1 *norm regularizer* $\lambda\|\mathbf{w}\|_1$ *is used in the well-known Lasso method for linear regression. Its proximal mapping promotes sparsity in the solution by setting some entries to zero if the corresponding component of* $\bar{\mathbf{w}}$ *is smaller than* $\eta\lambda$ *in magnitude.*

Example 3.5 (Group Lasso)**.** *This is an extension of Lasso that groups features together and enforces group-wise sparsity. Specifically, if one weight within a group is set to zero, then all weights in that group are simultaneously set to zero.*

Example 3.6 (Trace norm)**.** *The trace norm regularizer for a matrix is analogous to the ℓ_1 norm for a vector, as it promotes low-rank structure. Its proximal mapping induces a low-rank solution by setting the singular values of the input matrix to zero whenever they are smaller than $\eta\lambda$.*

3.2.1 Convex Functions

Lemma 3.6 *Consider the update*

$$\mathbf{w}_{t+1} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \mathbf{z}_t^\top(\mathbf{w} - \mathbf{w}_t) + \frac{1}{2\eta_t}\|\mathbf{w} - \mathbf{w}_t\|_2^2 + r(\mathbf{w}). \tag{3.22}$$

If r is μ_r-strongly convex, for any $\mathbf{w}$ we have

$$\mathbf{z}_t^\top(\mathbf{w}_{t+1} - \mathbf{w}) + r(\mathbf{w}_{t+1}) - r(\mathbf{w}) \le \frac{1}{2\eta_t}\|\mathbf{w}_t - \mathbf{w}\|_2^2 - (\frac{1}{2\eta_t} + \frac{\mu_r}{2})\|\mathbf{w}_{t+1} - \mathbf{w}\|_2^2 - \frac{1}{2\eta_t}\|\mathbf{w}_t - \mathbf{w}_{t+1}\|_2^2.$$

Proof. By the first-order optimality condition of (3.22), for any $\mathbf{w}$ we have

$$(\mathbf{z}_t + \partial r(\mathbf{w}_{t+1}) + \frac{1}{\eta_t}(\mathbf{w}_{t+1} - \mathbf{w}_t))^\top(\mathbf{w} - \mathbf{w}_{t+1}) \ge 0. \tag{3.23}$$

By the strong convexity of r, we have

$$r(\mathbf{w}_{t+1}) \le r(\mathbf{w}) + \partial r(\mathbf{w}_{t+1})^\top(\mathbf{w}_{t+1} - \mathbf{w}) - \frac{\mu_r}{2}\|\mathbf{w} - \mathbf{w}_{t+1}\|_2^2.$$

Adding the above two inequalities, we have

$$\mathbf{z}_t^\top(\mathbf{w}_{t+1} - \mathbf{w}) + r(\mathbf{w}_{t+1}) - r(\mathbf{w}) \le \frac{1}{\eta_t}(\mathbf{w}_t - \mathbf{w}_{t+1})^\top(\mathbf{w}_{t+1} - \mathbf{w}) - \frac{\mu_r}{2}\|\mathbf{w} - \mathbf{w}_{t+1}\|_2^2$$
$$= \frac{1}{2\eta_t}(\|\mathbf{w}_t - \mathbf{w}\|_2^2 - \|\mathbf{w}_{t+1} - \mathbf{w}\|_2^2 - \|\mathbf{w}_t - \mathbf{w}_{t+1}\|_2^2) - \frac{\mu_r}{2}\|\mathbf{w} - \mathbf{w}_{t+1}\|_2^2.$$

where the last equality uses the fact that $2(a-b)^\top(b-c) = \|a-c\|_2^2 - \|a-b\|_2^2 - \|b-c\|_2^2$. □

Theorem 3.5 *Suppose Assumption 3.6 holds. Let $\eta_t = \eta \le 1/L$ and $\bar{\mathbf{w}}_T = \frac{1}{T}\sum_{t=1}^T \mathbf{w}_{t+1}$. Then after T iterations of SPGD update (3.19), we have*

$$\mathbb{E}[F(\bar{\mathbf{w}}_T) - F(\mathbf{w}_*)] \leq \frac{\|\mathbf{w}_1 - \mathbf{w}_*\|_2^2}{2\eta T} + \eta\sigma^2.$$

If $\eta = \min(\frac{1}{L}, \frac{\|\mathbf{w}_1 - \mathbf{w}_*\|_2}{\sqrt{2T}\sigma})$, *then*

$$\mathbb{E}\left[F(\bar{\mathbf{w}}_T) - F(\mathbf{w}_*)\right] \leq \frac{\sqrt{2}\sigma\|\mathbf{w}_1 - \mathbf{w}_*\|_2}{\sqrt{T}} + \frac{L\|\mathbf{w}_1 - \mathbf{w}_*\|_2^2}{T}.$$

Why it matters

Insight 1: The theorem indicates that even if the objective has a non-smooth regularizer r, the convergence rate of SPGD still depends on the variance bound σ^2 instead of the Lipschitz constant of the objective function as in the analysis of SGD for non-smooth convex functions.
Insight 2: Employing the proximal mapping of r renders the convergence independent of the smoothness of r. Consequently, this approach is advantageous even when r is smooth, particularly if it possesses a large smoothness constant.

Proof. Without loss of generality, we assume g is μ-strongly convex with $\mu \geq 0$ and r is μ_r-strongly convex with $\mu_r \geq 0$ so that it covers both convex and strongly convex cases.

By Lemma 3.6, we have

$$\nabla g(\mathbf{w}_t, \zeta_t)^\top(\mathbf{w}_{t+1} - \mathbf{w}) + r(\mathbf{w}_{t+1}) \leq r(\mathbf{w}) + \frac{1}{2\eta_t}(\|\mathbf{w}_t - \mathbf{w}\|_2^2 - \|\mathbf{w}_{t+1} - \mathbf{w}\|_2^2)$$
$$- \frac{\mu_r}{2}\|\mathbf{w} - \mathbf{w}_{t+1}\|_2^2 - \frac{1}{2\eta_t}\|\mathbf{w}_t - \mathbf{w}_{t+1}\|_2^2.$$

By the smoothness of g, we have

$$g(\mathbf{w}_{t+1}) \leq g(\mathbf{w}_t) + \nabla g(\mathbf{w}_t)^\top(\mathbf{w}_{t+1} - \mathbf{w}_t) + \frac{L}{2}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2.$$

By the strong convexity of g, we have

$$g(\mathbf{w}_t) \leq g(\mathbf{w}) + \nabla g(\mathbf{w}_t)^\top(\mathbf{w}_t - \mathbf{w}) - \frac{\mu}{2}\|\mathbf{w}_t - \mathbf{w}\|_2^2.$$

Adding the above two inequalities, we have

$$g(\mathbf{w}_{t+1}) \leq g(\mathbf{w}) + \nabla g(\mathbf{w}_t)^\top(\mathbf{w}_{t+1} - \mathbf{w}) - \frac{\mu}{2}\|\mathbf{w}_t - \mathbf{w}\|_2^2 + \frac{L}{2}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2.$$

Combining this with the first inequality for $\mathbf{w} = \mathbf{w}_*$, we have

$$
\begin{aligned}
F(\mathbf{w}_{t+1}) - F(\mathbf{w}_*) \leq{}& \frac{1}{2\eta_t}(\|\mathbf{w}_t - \mathbf{w}_*\|_2^2 - \|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2^2 - \|\mathbf{w}_t - \mathbf{w}_{t+1}\|_2^2) \\
&- \frac{\mu}{2}\|\mathbf{w}_t - \mathbf{w}_*\|_2^2 - \frac{\mu_r}{2}\|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2^2 + \frac{L}{2}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 \\
&+ (\nabla g(\mathbf{w}_t) - \nabla g(\mathbf{w}_t, \zeta_t))^\top (\mathbf{w}_{t+1} - \mathbf{w}_*).
\end{aligned} \tag{3.24}
$$

This is similar to (3.8) except for the two negative terms $-\frac{\mu}{2}\|\mathbf{w}_t - \mathbf{w}_*\|_2^2 - \frac{\mu_r}{2}\|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2^2$, which are due to the μ_r-strong convexity of r and μ-strong convexity of g. If $\mu_r = \mu = 0$, the remaining proof is similar to that of Theorem 3.1 with the following definition of $\hat{\mathbf{w}}_{t+1}$:

$$
\hat{\mathbf{w}}_{t+1} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \frac{1}{2\eta_t}\|\mathbf{w} - (\mathbf{w}_t - \eta_t \nabla g(\mathbf{w}_t))\|_2^2 + r(\mathbf{w}).
$$

It used to bound the expectation of last term in the RHS of (3.24):

$$
\begin{aligned}
&\mathbb{E}[(\nabla g(\mathbf{w}_t) - g(\mathbf{w}_t, \zeta_t))^\top (\mathbf{w}_{t+1} - \hat{\mathbf{w}}_{t+1} + \hat{\mathbf{w}}_{t+1} - \mathbf{w}_*)] \\
&= \mathbb{E}[(\nabla g(\mathbf{w}_t) - g(\mathbf{w}_t, \zeta_t))^\top (\mathbf{w}_{t+1} - \hat{\mathbf{w}}_{t+1})] \leq \eta_t \mathbb{E}[\|(\nabla g(\mathbf{w}_t) - g(\mathbf{w}_t, \zeta_t)\|_2^2] = \eta_t \sigma^2,
\end{aligned} \tag{3.25}
$$

where the inequality is due to Lemma 1.7.

□

3.2.2 Strongly Convex Functions

We can prove a faster convergence when the loss function or the regularizer is strongly convex.

Theorem 3.6 *Suppose Assumption 3.6 holds and g is μ-strongly convex and r is μ_r-strongly convex. Let $\eta_t = 1/((\mu + \mu_r)t + L)$ and $\bar{\mathbf{w}}_T = \frac{1}{T}\sum_{t=1}^T \mathbf{w}_{t+1}$. Then after T iterations of SPGD update (3.19), we have*

$$
\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^T (F(\mathbf{w}_{t+1}) - F(\mathbf{w}_*))\right] \leq \frac{(L + \mu_r)\,\|\mathbf{w}_1 - \mathbf{w}_*\|_2^2}{T} + \frac{(1 + \log T)\sigma^2}{T(\mu + \mu_r)}.
$$

Proof. Similar to the proof of Theorem 3.5, if $\eta_t \leq \frac{1}{L}$ we have

$$
\begin{aligned}
&\mathbb{E}[(F(\mathbf{w}_{t+1}) - F(\mathbf{w}_*))] \\
&\leq \mathbb{E}\left[(\frac{1}{2\eta_t}\|\mathbf{w}_t - \mathbf{w}_*\|_2^2 - \frac{1}{2\eta_t}\|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2^2 - \frac{\mu}{2}\|\mathbf{w}_t - \mathbf{w}_*\|_2^2 - \frac{\mu_r}{2}\|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2^2\right] \\
&+ \eta_t \sigma^2.
\end{aligned}
$$

Taking summation over $t = 1, \ldots, T$ we have

Algorithm 3 `Restarted SPGD`

1: **Input:** a learning schedule $\{\eta_k, T_K\}_{k=1}^T$, starting point $\mathbf{w}_1$
2: **for** $k = 1, \ldots, K$ **do**
3: run SPGD with a learning rate η_k for T_k iterations starting from $\mathbf{w}_k$
4: return an averaged solution $\mathbf{w}_{k+1}$
5: **end for**

$$\mathbb{E}\left[\sum_{t=1}^{T}(F(\mathbf{w}_{t+1}) - F(\mathbf{w}_*))\right]$$
$$\leq \mathbb{E}\left[\sum_{t=1}^{T}(\frac{1}{2\eta_t} - \frac{1}{2\eta_{t-1}} - \frac{\mu+\mu_r}{2})\|\mathbf{w}_t - \mathbf{w}_*\|_2^2 + \frac{1}{2\eta_0}\|\mathbf{w}_1 - \mathbf{w}_*\|_2^2 + \frac{\mu_r}{2}\|\mathbf{w}_1 - \mathbf{w}_*\|_2^2\right]$$
$$+ \sum_{t=1}^{T}\eta_t\sigma^2.$$

Let $\eta_t = \frac{1}{(\mu+\mu_r)t+L}$. Then $\frac{1}{2\eta_t} - \frac{1}{2\eta_{t-1}} - \frac{\mu+\mu_r}{2} = 0$ and we have

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}(F(\mathbf{w}_{t+1}) - F(\mathbf{w}_*))\right]$$
$$\leq \frac{L+\mu_r}{2T}\|\mathbf{w}_1 - \mathbf{w}_*\|_2^2 + \frac{1}{T}\sum_{t=1}^{T}\frac{\sigma^2}{(\mu+\mu_r)t} \leq \frac{L+\mu_r}{2T}\|\mathbf{w}_1 - \mathbf{w}_*\|_2^2 + \frac{(1+\log T)\sigma^2}{T(\mu+\mu_r)}.$$

□

A Restarted Approach

The $\log T$ factor in the convergence bound can be removed using a restarting scheme. It runs in multiple stages. At stage k, it start with a step size η_k and ran SGD with a number of iterations T_k and returns an averaged solution $\mathbf{w}_k$. By choosing η_k, T_k appropriately, after a logarithmic number of K stages, we will get a solution $\mathbf{w}_K$ satisfying $\mathbb{E}[F(\mathbf{w}_K) - F(\mathbf{w}_*)] \leq \epsilon$. The key motivation is coming from the one-stage convergence bound in Theorem 3.5:

$$\mathbb{E}[F(\bar{\mathbf{w}}_T) - F(\mathbf{w}_*)] \leq \frac{\|\mathbf{w}_1 - \mathbf{w}_*\|_2^2}{2\eta T} + \eta\sigma^2. \tag{3.26}$$

Since the μ-strong convexity of F implies that $\|\mathbf{w}_1 - \mathbf{w}_*\|_2^2 \leq \frac{2}{\mu}(F(\mathbf{w}_1) - F(\mathbf{w}_*))$, then we can establish a recursion of the objective gap in a stage-wise manner. From which, we can show the objective gap will decrease geometrically if η_k decreases

geometrically and T_k increases accordingly. This is formally stated in the following theorem.

Theorem 3.7 *Suppose Assumption 3.6 holds, F is μ-strongly convex and there exists ϵ_1 such that $F(\mathbf{w}_1) - F(\mathbf{w}_*) \leq \epsilon_1$. Let $\eta_k = \min(\frac{1}{L}, \frac{\epsilon_1}{2^{k+1}\sigma^2})$ and $T_k = \frac{4}{\mu\eta_k}$. Then after $K = \lfloor \log_2(\epsilon_1/\epsilon) \rfloor$ stages of Restarted SPGD updates (Alg. 3), we have*

$$\mathbb{E}\left[F(\mathbf{w}_{K+1}) - F(\mathbf{w}_*))\right] \leq \epsilon.$$

The iteration complexity is $\sum_{k=1}^{K} T_k = O(\frac{\sigma^2}{\mu\epsilon} + \frac{L}{\mu}\log(\frac{\epsilon_1}{\epsilon}))$.

Proof. Let $\epsilon_k = \epsilon_1/2^k$. Then $\epsilon_{K+1} = \epsilon_1/2^{K+1} \leq \epsilon$ and $\epsilon_K \geq \epsilon$.

Applying the one-stage analysis of SPGD, we have

$$\mathbb{E}[F(\bar{\mathbf{w}}_{k+1}) - F(\mathbf{w}_*)] \leq \frac{\mathbb{E}[\|\mathbf{w}_k - \mathbf{w}_*\|_2^2]}{2\eta_k T_k} + \eta_k\sigma^2 \leq \frac{\mathbb{E}[F(\mathbf{w}_k) - F(\mathbf{w}_*)]}{\mu\eta_k T_k} + \eta_k\sigma^2.$$

Then we prove by induction. Assume $\mathbb{E}[F(\mathbf{w}_k) - F(\mathbf{w}_*)] \leq \epsilon_k$, we prove $\mathbb{E}[F(\mathbf{w}_{k+1}) - F(\mathbf{w}_*)] \leq \epsilon_{k+1}$.

$$\begin{aligned}\mathbb{E}[F(\bar{\mathbf{w}}_{k+1}) - F(\mathbf{w}_*)] &\leq \frac{\mathbb{E}[\|\mathbf{w}_k - \mathbf{w}_*\|_2^2]}{2\eta_k T_k} + \eta_k\sigma^2. \\ &\leq \frac{\epsilon_k}{\mu\eta_k T_k} + \eta_k\sigma^2 \leq \frac{\epsilon_k}{\mu\eta_k T_k} + \frac{\epsilon_{k+1}}{2} \leq \frac{\epsilon_k}{4} + \frac{\epsilon_{k+1}}{2} = \epsilon_{k+1}.\end{aligned}$$

Thus, $\mathbb{E}[F(\mathbf{w}_{K+1}) - F(\mathbf{w}_*)] \leq \epsilon_{K+1} \leq \epsilon$. The total number of iterations is

$$\begin{aligned}\sum_{k=1}^{K} T_k &= \sum_{k=1}^{K} \frac{4}{\mu\eta_k} = \sum_{k=1}^{K} \max\left(\frac{4\cdot 2^{k+1}\sigma^2}{\mu\epsilon_1}, \frac{4L}{\mu}\right) \\ &\leq \sum_{k=1}^{K} \max\left(\frac{8\sigma^2}{\mu\epsilon 2^{K-k}}, \frac{4L}{\mu}\right) = O\left(\frac{\sigma^2}{\mu\epsilon} + \frac{L}{\mu}\log\left(\frac{\epsilon_1}{\epsilon}\right)\right).\end{aligned}$$

□

Last-iterate Convergence

Furthermore, if $g(\cdot)$ and/or r is strongly convex, we can also prove $\|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2$ converges to zero.

Lemma 3.7 *If g is L-smooth and μ-strongly convex and r is μ_r-strongly convex, for the update (3.19) with $\eta_t \leq 2/L$ we have*

$$\mathbb{E}_{\zeta_t}[\|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2^2] \leq \frac{(1 - (2\eta_t - \eta_t^2 L)\mu)\|\mathbf{w}_t - \mathbf{w}_*\|_2^2 + \eta_t^2\sigma^2}{1 + \eta\mu_r}. \tag{3.27}$$

If g μ-strongly convex and $\|\partial g(\mathbf{w})\|_2 \le G$ for $\mathbf{w} \in dom(r)$, for the update (3.19) we have

$$\mathbb{E}_{\zeta_t}[\|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2^2] \le \frac{(1 - 2\eta_t\mu)\|\mathbf{w}_t - \mathbf{w}_*\|_2^2 + \eta_t^2(\sigma^2 + 4G^2)}{1 + \eta\mu_r}. \tag{3.28}$$

Proof. Let $\mathbb{E}_t = \mathbb{E}_{\zeta_t}$. Let us consider smooth case first. Due to the optimality condition of $\mathbf{w}_*$, we have

$$\begin{aligned}\mathbf{w}_* &= \arg\min_{\mathbf{w}\in\mathbb{R}^d} \nabla g(\mathbf{w}_*)^\top(\mathbf{w} - \mathbf{w}_*) + \frac{1}{2\eta_t}\|\mathbf{w} - \mathbf{w}_*\|_2^2 + r(\mathbf{w}) \\ &= \text{prox}_{\eta_t r}(\mathbf{w}_* - \eta_t \nabla g(\mathbf{w}_*)).\end{aligned}$$

Due to the Lipschitz continuity of the prox operator (see Lemma 1.7), we have

$$\mathbb{E}_t\|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2^2 \le \frac{1}{1 + \eta\mu_r}\mathbb{E}_t\|\mathbf{w}_t - \eta_t\nabla g(\mathbf{w}_t; \zeta_t) - [\mathbf{w}_* - \eta_t\nabla g(\mathbf{w}_*)]\|_2^2. \tag{3.29}$$

Next, we bound

$$\begin{aligned}&\mathbb{E}_t\|\mathbf{w}_t - \eta_t\nabla g(\mathbf{w}_t; \zeta_t) - [\mathbf{w}_* - \eta_t\nabla g(\mathbf{w}_*)]\|_2^2 \\ &= \mathbb{E}_t\|[\mathbf{w}_t - \eta_t\nabla g(\mathbf{w}_t)] - [\mathbf{w}_* - \eta_t\nabla g(\mathbf{w}_*)] + \eta_t\nabla g(\mathbf{w}_t) - \eta_t\nabla g(\mathbf{w}_t, \zeta_t)\|_2^2 \\ &= \mathbb{E}_t\|[\mathbf{w}_t - \eta_t\nabla g(\mathbf{w}_t)] - [\mathbf{w}_* - \eta_t\nabla g(\mathbf{w}_*)]\|_2^2 + \eta_t^2\sigma^2,\end{aligned}$$

where the last inequality uses $\mathbb{E}_t[\nabla g(\mathbf{w}_t, \zeta_t) - \nabla g(\mathbf{w}_t)] = 0$ by expanding the RHS. Let us bound the first term below.

$$\begin{aligned}&\mathbb{E}_t\|[\mathbf{w}_t - \eta_t\nabla g(\mathbf{w}_t)] - [\mathbf{w}_* - \eta_t\nabla g(\mathbf{w}_*)]\|_2^2 \\ &= \mathbb{E}_t\|\mathbf{w}_t - \mathbf{w}_*\|_2^2 + \eta_t^2\mathbb{E}_t\|\nabla g(\mathbf{w}_t) - \nabla g(\mathbf{w}_*)\|_2^2 - 2\eta_t\mathbb{E}_t(\mathbf{w}_t - \mathbf{w}_*)^\top(\nabla g(\mathbf{w}_t) - \nabla g(\mathbf{w}_*)) \\ &\le \mathbb{E}_t\|\mathbf{w}_t - \mathbf{w}_*\|_2^2 + \eta_t^2 L\mathbb{E}_t(\mathbf{w}_t - \mathbf{w}_*)^\top(\nabla g(\mathbf{w}_t) - \nabla g(\mathbf{w}_*)) \\ &- 2\eta_t\mathbb{E}_t(\mathbf{w}_t - \mathbf{w}_*)^\top(\nabla g(\mathbf{w}_t) - \nabla g(\mathbf{w}_*)) \\ &= \mathbb{E}_t\|\mathbf{w}_t - \mathbf{w}_*\|_2^2 - (2\eta_t - \eta_t^2 L)\mathbb{E}_t(\mathbf{w}_t - \mathbf{w}_*)^\top(\nabla g(\mathbf{w}_t) - \nabla g(\mathbf{w}_*)) \\ &\le \mathbb{E}_t\|\mathbf{w}_t - \mathbf{w}_*\|_2^2 - (2\eta_t - \eta_t^2 L)\mu\mathbb{E}_t\|\mathbf{w}_t - \mathbf{w}_*\|_2^2 \\ &\le (1 - (2\eta_t - \eta_t^2 L)\mu)\mathbb{E}_t\|\mathbf{w}_t - \mathbf{w}_*\|_2^2,\end{aligned}$$

where the first inequality uses Lemma 1.5(c) and the second inequality follows from Lemma 1.6(c).

If g is non-smooth, we bound $\mathbb{E}\|\nabla g(\mathbf{w}_t) - \nabla g(\mathbf{w}_*)\|_2^2 \le 4G^2$. Combining this with (3.29) concludes the proof.

□

Theorem 3.8 *Suppose Assumption 3.6 holds and g is μ-strongly convex and r is μ_r-strongly convex. Let $\eta_t = \eta \le \min(1/L, 1/\mu_r)$. Then after T iterations of SPGD (3.19) update, we have*

$$\mathbb{E}[\|\mathbf{w}_{T+1} - \mathbf{w}_*\|_2^2] \leq e^{-\frac{\eta(\mu+\mu_r)T}{2}}\mathbb{E}[\|\mathbf{w}_1 - \mathbf{w}_*\|_2^2] + \frac{\eta\sigma^2}{\mu+\mu_r}. \tag{3.30}$$

Why it matters

This theorem indicates that if we set $\eta \leq O((\mu+\mu_r)\epsilon/\sigma^2)$, then with $T = \tilde{O}\left(\frac{\sigma^2}{(\mu+\mu_r)^2\epsilon}\right)$ iterations, the algorithm finds an solution $\mathbf{w}_{T+1}$ that is ϵ-close to the optimal solution $\mathbf{w}_*$ measured by $\mathbb{E}[\|\mathbf{w}_{T+1} - \mathbf{w}_*\|_2^2]$, where $\tilde{O}(\cdot)$ hides a logarithmic factor of $\log(1/\epsilon)$.

Proof. If $\eta \leq 1/L$, Lemma 3.7 implies that

$$\begin{aligned}
\mathbb{E}[\|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2^2] &\leq \frac{(1-\eta\mu)\mathbb{E}[\|\mathbf{w}_t - \mathbf{w}_*\|_2^2] + \eta^2\sigma^2}{1+\eta\mu_r} \\
&\leq \left(1 - \frac{\eta\mu_r}{2}\right)\{(1-\eta\mu)\mathbb{E}[\|\mathbf{w}_t - \mathbf{w}_*\|_2^2] + \eta^2\sigma^2\} \\
&\leq \left(1 - \frac{\eta\mu_r}{2} - \eta\mu + \frac{\eta^2\mu\mu_r}{2}\right)\mathbb{E}[\|\mathbf{w}_t - \mathbf{w}_*\|_2^2] + \eta^2\sigma^2,
\end{aligned}$$

where the first inequality is due to $1 \leq (1+\eta\mu_r)(1-\frac{\eta\mu_r}{2}) = 1 + \frac{\eta\mu_r}{2} - \frac{\eta^2\mu_r^2}{2}$ as $\eta\mu_r \leq 1$. Then

$$\mathbb{E}[\|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2^2] \leq (1 - \frac{\eta\mu_r}{2} - \frac{\eta\mu}{2})\mathbb{E}[\|\mathbf{w}_t - \mathbf{w}_*\|_2^2] + \eta^2\sigma^2.$$

Unroll this inequality for $t = 1, \ldots, T$, we have

$$\mathbb{E}[\|\mathbf{w}_{T+1} - \mathbf{w}_*\|_2^2] \leq \left(1 - \frac{\eta(\mu+\mu_r)}{2}\right)\mathbb{E}\|\mathbf{w}_T - \mathbf{w}_*\|_2^2 + \eta^2\sigma^2.$$

Applying this inequality T times gives

$$\begin{aligned}
&\mathbb{E}[\|\mathbf{w}_{T+1} - \mathbf{w}_*\|_2^2] \\
&\leq \left(1 - \frac{\eta(\mu+\mu_r)}{2}\right)^T \mathbb{E}\|\mathbf{w}_1 - \mathbf{w}_*\|_2^2 + \sum_{t=0}^{T-1}\left(1 - \frac{\eta(\mu+\mu_r)}{2}\right)^t \eta^2\sigma^2.
\end{aligned}$$

Since $(1-\alpha)^T \leq e^{-\alpha T}$ for $\alpha \in (0,1)$ and $\sum_{t=0}^{T-1}\alpha^t < \frac{1}{1-\alpha}$, we have

$$\begin{aligned}
&\mathbb{E}[\|\mathbf{w}_{T+1} - \mathbf{w}_*\|_2^2] \leq e^{-\frac{\eta(\mu+\mu_r)T}{2}}\mathbb{E}[\|\mathbf{w}_1 - \mathbf{w}_*\|_2^2] + \eta^2\sigma^2\frac{2}{\eta(\mu+\mu_r)} \\
&= e^{-\frac{\eta(\mu+\mu_r)T}{2}}\mathbb{E}[\|\mathbf{w}_1 - \mathbf{w}_*\|_2^2] + \frac{2\eta\sigma^2}{\mu+\mu_r}.
\end{aligned}$$

□

Corollary 3.1. *Under the setting of Theorem 3.8, if* $\frac{1}{\eta_t} = \frac{\bar{\mu}}{2} + \sqrt{(\frac{\bar{\mu}}{2})^2 + \frac{1}{\eta_{t-1}^2}}$ *with* $\eta_0 \leq \min(1/L, 1/\mu_r)$ *and* $\bar{\mu} = (\mu + \mu_r)/2$*, then we have*

$$\mathbb{E}[\|\mathbf{w}_{T+1} - \mathbf{w}_*\|_2^2] \leq \frac{4\|\mathbf{w}_1 - \mathbf{w}_*\|_2^2}{\eta_0^2 \bar{\mu}^2 T^2} + \frac{2\sigma^2}{\bar{\mu}^2 T}.$$

Proof. Let $\delta_t = \|\mathbf{w}_t - \mathbf{w}_*\|_2^2$. Due to the update of η_t, we have $1 - \bar{\mu}\eta_t = \frac{\eta_t^2}{\eta_{t-1}^2}$. Hence, we have:

$$\mathbb{E}[\ \delta_{T+1}] \leq \mathbb{E}[(1 - \bar{\mu}\eta_T)\delta_T] + \sigma^2\eta_T^2 \leq \mathbb{E}\left[\frac{\eta_T^2}{\eta_{T-1}^2}\delta_T\right] + \sigma^2\eta_T^2.$$

Unrolling this inequality for $t = 1, \ldots, T$, we have

$$\mathbb{E}[\ \delta_{T+1}] \leq \mathbb{E}\left[\frac{\eta_T^2}{\eta_{T-2}^2}\delta_{T-1}\right] + \sigma^2\eta_T^2 * 2 \leq \frac{\eta_T^2}{\eta_0^2}\delta_1 + \sigma^2\eta_T^2 * T.$$

Since $\frac{1}{\eta_t} = \frac{\bar{\mu}}{2} + \sqrt{(\frac{\bar{\mu}}{2})^2 + \frac{1}{\eta_{t-1}^2}}$. Then, we have $\frac{1}{\eta_t} \geq \frac{\bar{\mu}}{2} + \frac{1}{\eta_{t-1}}$. As a result, $\frac{1}{\eta_T} \geq \frac{\bar{\mu}T}{2} + \frac{1}{\eta_0} \geq \max(L, \mu_r)$, where $\eta_0 \leq \min(\frac{1}{L}, \frac{1}{\mu_r})$. Hence, $\eta_T \leq \frac{2}{\bar{\mu}T}$, and

$$\mathbb{E}[\delta_{T+1}] \leq \frac{4\delta_1}{\eta_0^2 \bar{\mu}^2 T^2} + \frac{2\sigma^2}{\bar{\mu}^2 T}.$$

□

Why it matters

This corollary shows that a decreasing learning rate schedule can be used without requiring prior knowledge of ϵ, in order to obtain a solution $\mathbf{w}_{T+1}$ that is ϵ-close to the optimum $\mathbf{w}_*$, measured by $\mathbb{E}[\|\mathbf{w}_{T+1} - \mathbf{w}_*\|_2^2]$. The iteration complexity is

$$T = O\left(\max\left\{\frac{1}{\bar{\mu}\eta_0\sqrt{\epsilon}}, \frac{\sigma^2}{\bar{\mu}^2\epsilon}\right\}\right).$$

3.3 Stochastic Coordinate Descent

In this section, we present stochastic coordinate descent (SCD) for solving the stochastic optimization:

$$\min_{\alpha \in \Omega \subseteq \mathbb{R}^n} f(\alpha) = \mathbb{E}[f(\alpha, \xi)]. \tag{3.31}$$

where $\Omega = \Omega_1 \times \Omega_2 \cdots \times \Omega_n$.

The key motivation is that if the dimensionality n of α is very large, then computing $\nabla f(\alpha, \xi)$ could be expensive at each iteration. However, if the function exhibits decomposable structure over dimensions of α, then we can sample a random coordinate of α and update it. To this end, we assume that $[\nabla f(\alpha, \xi)]_i, \forall i \in [n]$ is easy to compute. In machine learning applications, this is possible if $f(\alpha, \xi) = \alpha^\top \boldsymbol{g}(\xi)$ and computing each coordinate of $\boldsymbol{g}(\xi)$ is much more cheaper than computing itself. An example is the COCE problem (2.62), which will be discussed in Section 5.5.

Let us consider a simple version of SCD. At each iteration t, a coordinate denoted by i_t is randomly sampled from $\{1, \ldots, n\}$ with uniform probabilities. Then we compute $\nabla_{i_t} f(\alpha_t, \xi_t) = [\nabla f(\alpha_t, \xi_t)]_{i_t}$ and update α by

$$\alpha_{t+1,i} = \begin{cases} \Pi_{\Omega_i}[\alpha_{t,i} - \eta \nabla_i f(\alpha_t, \xi_t)] & \text{if } i = i_t \\ \alpha_{t,i} & \text{o.w.} \end{cases}$$

Convergence Analysis

We make the following assumption.

Assumption 3.7. *The following conditions hold:*

(i) $f(\alpha)$ *is convex;*
(ii) For any α*, we have* $\mathbb{E}[\|\nabla_i f(\alpha; \zeta) - \nabla_i f(\alpha)\|_2^2] \le \sigma_i^2$ *for some* $\sigma_i \ge 0$*;*
(iii) ∇f *is* L_i*-Lipschitz continuous w.r.t to the i-th coordinate, i.e.,*

$$\|\nabla f(\alpha) - \nabla f(\alpha + \mathbf{e}_i \delta)\|_2 \le L_i |\delta|.$$

Theorem 3.9 *Let* $\bar{\alpha}_T = \frac{1}{T}\sum_{t=1}^T \alpha_{t+1}$, $\bar{L} = \max_i L_i$. *If* $\eta_t = \eta \le \frac{1}{\bar{L}}$*, after T iterations of SCD update we have*

$$\mathbb{E}\Big[f(\bar{\alpha}_t) - f(\alpha_*)\Big] \le \frac{(n-1)(f(\alpha_1) - f(\alpha_*))}{T} + \frac{n}{2\eta T}\|\alpha_1 - \alpha_*\|_2^2 + \sum_{i=1}^n \eta \sigma_i^2.$$

If $\|\alpha_1 - \alpha_*\|_2^2 \le D^2$, $\sum_{i=1}^n \sigma_i^2 \le \sigma^2$*, with* $\eta = O(\min(\frac{\sqrt{n}}{\sqrt{2T}\sigma}, 1/\bar{L}))$*, we have*

$$\mathbb{E}\Big[f(\bar{\alpha}_t) - f(\alpha_*)\Big] \le \frac{(n-1)(f(\alpha_1) - f(\alpha_*))}{T} + \frac{\sqrt{2n}D\sigma}{\sqrt{T}} + \frac{\bar{L}nD^2}{T}.$$

Why it matters

According to the theorem, SCD's iteration complexity is $O(\frac{nD^2\sigma^2}{\epsilon^2})$. Although this is n times higher than that of SGD, it is offset by the fact that each individual iteration of SCD can be n times cheaper to compute.

Algorithm 4 SCD

1: **Input:** learning rate schedule $\{\eta_t\}_{t=1}^T$, starting point α_1
2: **for** $t = 1, \dots, T$ **do**
3: Sample a coordinate i_t uniformly
4: Compute an unbiased coordinate gradient estimator $\nabla_{i_t} f(\alpha_t, \xi_t)$
5: Update

$$\alpha_{t+1,i} = \begin{cases} \Pi_{\Omega_i}[\alpha_{t,i} - \eta_t \nabla_i f(\alpha_t, \xi_t)] & \text{if } i = i_t \\ \alpha_{t,i} & \text{o.w.} \end{cases}$$

6: **end for**

Proof. To facilitate the analysis, we consider a virtual sequence $\{\tilde{\alpha}_t\}$ defined by

$$\tilde{\alpha}_{t+1} = \Pi_\Omega[\alpha_t - \eta_t \nabla f(\alpha_t, \xi_t)].$$

Due to the decomposability of $\Omega = \Omega_1 \times \cdots \Omega_n$, it implies that

$$\tilde{\alpha}_{t+1,i} = \Pi_{\Omega_i}[\alpha_{t,i} - \eta_t \nabla_i f(\alpha_t, \xi_t)], \forall i.$$

Applying Lemma 3.6 to each coordinate of $\tilde{\alpha}_{t+1}$ with $r(\alpha_i) = \mathbb{I}_{0-\infty}(\alpha_i \in \Omega_i)$, we have

$$\mathbb{E}[\nabla_i f(\alpha_t, \xi_t)^\top (\tilde{\alpha}_{t+1,i} - \alpha_{*,i})] \le \frac{1}{2\eta_t}\mathbb{E}[\|\alpha_{t,i} - \alpha_{*,i}\|_2^2 - \frac{1}{2\eta_t}\|\tilde{\alpha}_{t+1,i} - \alpha_{*,i}\|_2^2] - \frac{1}{2\eta_t}\mathbb{E}[\|\alpha_{t,i} - \tilde{\alpha}_{t+1,i}\|_2^2].$$

Then,

$$\mathbb{E}[\nabla_i f(\alpha_t)^\top (\tilde{\alpha}_{t+1,i} - \alpha_{*,i})] \le \frac{1}{2\eta_t}\mathbb{E}[\|\alpha_{t,i} - \alpha_{*,i}\|_2^2 - \frac{1}{2\eta_t}\|\tilde{\alpha}_{t+1,i} - \alpha_{*,i}\|_2^2] - \frac{1}{2\eta_t}\mathbb{E}[\|\alpha_{t,i} - \tilde{\alpha}_{t+1,i}\|_2^2] + \mathbb{E}[(\nabla_i f(\alpha_t) - \nabla_i f(\alpha_t, \xi_t))^\top (\tilde{\alpha}_{t+1,i} - \alpha_{*,i})].$$

Similar to (3.25), the last term in the RHS can be bounded by $\mathbb{E}[(\nabla_i f(\alpha_t) - \nabla_i f(\alpha_t, \xi_t))^\top (\tilde{\alpha}_{t+1,i} - \alpha_{*,i})] \le \mathbb{E}(\nabla_i f(\alpha_t) - \nabla_i f(\alpha_t, \xi_t))^2 \le \eta_t \sigma_i^2$. Then adding the above inequality over $i = 1, \dots, n$, we have

$$\mathbb{E}[\nabla_i f(\alpha_t)^\top (\tilde{\alpha}_{t+1,i} - \alpha_{*,i})] \le \frac{1}{2\eta_t}\mathbb{E}\left[\|\alpha_{t,i} - \alpha_{*,i}\|_2^2 - \frac{1}{2\eta_t}\|\tilde{\alpha}_{t+1,i} - \alpha_{*,i}\|_2^2\right] - \frac{1}{2\eta_t}\mathbb{E}[\|\alpha_{t,i} - \tilde{\alpha}_{t+1,i}\|_2^2] + \eta_t \sigma_i^2.$$

Due to the randomness of i_t, we have

$$\mathbb{E}[(\alpha_{t+1,i} - \alpha_{*,i})^2] = \frac{1}{n}\mathbb{E}[(\tilde{\alpha}_{t+1,i} - \alpha_{*,i})^2] + (1 - \frac{1}{n})\mathbb{E}[(\alpha_{t,i} - \alpha_{*,i})^2]$$
$$\mathbb{E}[\nabla_i f(\alpha_t)^\top(\alpha_{t+1,i} - \alpha_{*,i})] = \frac{1}{n}\mathbb{E}[\nabla_i f(\alpha_t)^\top(\tilde{\alpha}_{t+1,i} - \alpha_{*,i})]$$
$$+ (1 - \frac{1}{n})\mathbb{E}[\nabla_i f(\alpha_t)^\top(\alpha_{t,i} - \alpha_{*,i})]$$
$$\mathbb{E}[\|\alpha_{t,i} - \alpha_{t+1,i}\|_2^2] = \frac{1}{n}\mathbb{E}[\|\alpha_{t,i} - \tilde{\alpha}_{t+1,i}\|_2^2].$$

Combining the above, we have

$$\mathbb{E}[n\nabla_i f(\alpha_t)^\top(\alpha_{t+1,i} - \alpha_{*,i}) - (n-1)\nabla_i f(\alpha_t)^\top(\alpha_{t,i} - \alpha_{*,i})]$$
$$\leq \frac{1}{2\eta_t}\mathbb{E}[\|\alpha_{t,i} - \alpha_{*,i}\|_2^2] - \frac{1}{2\eta_t}\mathbb{E}[(n\|\alpha_{t+1,i} - \alpha_{*,i}\|_2^2 - (n-1)\|\alpha_{t,i} - \alpha_{*,i}\|_2^2)]$$
$$- \frac{n}{2\eta_t}\mathbb{E}[\|\alpha_{t,i} - \alpha_{t+1,i}\|_2^2] + \eta_t\sigma_i^2.$$

Adding this over $i = 1, \ldots, n$, we have

$$\mathbb{E}\left[n\sum_{i=1}^n \nabla_i f(\alpha_t)^\top(\alpha_{t+1,i} - \alpha_{*,i}) - \sum_{i=1}^n (n-1)\nabla_i f(\alpha_t)^\top(\alpha_{t,i} - \alpha_{*,i})\right]$$
$$\leq \frac{n}{2\eta_t}\mathbb{E}\left[\sum_{i=1}^n \|\alpha_{t,i} - \alpha_{*,i}\|_2^2\right] - \frac{n}{2\eta_t}\mathbb{E}\left[\sum_{i=1}^n \|\alpha_{t+1,i} - \alpha_{*,i}\|_2^2\right]$$
$$- \frac{n}{2\eta_t}\mathbb{E}\left[\sum_{i=1}^n \|\alpha_{t,i} - \alpha_{t+1,i}\|_2^2\right] + \sum_{i=1}^n \eta_t\sigma_i^2.$$

For the LHS, we have

$$n\sum_{i=1}^n \nabla_i f(\alpha_t)^\top(\alpha_{t+1,i} - \alpha_{*,i}) - \sum_{i=1}^n (n-1)\nabla_i f(\alpha_t)^\top(\alpha_{t,i} - \alpha_{*,i})$$
$$= n\nabla_{i_t} f(\alpha_t)^\top(\alpha_{t+1,i_t} - \alpha_{*,i_t}) + n\sum_{i\neq i_t}^n \nabla_i f(\alpha_t)^\top(\alpha_{t,i} - \alpha_{*,i})$$
$$- \sum_{i=1}^n (n-1)\nabla_i f(\alpha_t)^\top(\alpha_{t,i} - \alpha_{*,i})$$
$$= n\nabla_{i_t} f(\alpha_t)^\top(\alpha_{t+1,i_t} - \alpha_{*,i_t}) - n\nabla_{i_t} f(\alpha_t)^\top(\alpha_{t,i_t} - \alpha_{*,i_t})$$
$$+ \sum_{i=1}^n \nabla_i f(\alpha_t)^\top(\alpha_{t,i} - \alpha_{*,i})$$
$$= n\nabla_{i_t} f(\alpha_t)^\top(\alpha_{t+1,i_t} - \alpha_{t,i_t}) + \nabla f(\alpha_t)^\top(\alpha_t - \alpha_*).$$

By the assumption, we have

$$\begin{aligned}\nabla_{i_t} f(\alpha_t)^\top (\alpha_{t+1,i_t} - \alpha_{t,i_t}) &= \nabla f(\alpha_t)^\top \mathbf{e}_{i_t} (\alpha_{t+1,i_t} - \alpha_{t,i_t}) \\ &\geq f(\alpha_{t+1}) - f(\alpha_t) - \frac{L_{i_t}}{2} \|(\alpha_{t+1,i_t} - \alpha_{t,i_t})\|_2^2 \\ \nabla f(\alpha_t)^\top (\alpha_t - \alpha_*) &\geq f(\alpha_t) - f(\alpha_*).\end{aligned}$$

Combining the above, we have

$$\begin{aligned}&n \sum_{i=1}^n \nabla_i f(\alpha_t)^\top (\alpha_{t+1,i} - \alpha_{*,i}) - \sum_{i=1}^n (n-1) \nabla_i f(\alpha_t)^\top (\alpha_{t,i} - \alpha_{*,i}) \\ &\geq n(f(\alpha_{t+1}) - f(\alpha_t) - \frac{L_{i_t}}{2} \|\alpha_{t+1,i_t} - \alpha_{t,i_t}\|_2^2) + f(\alpha_t) - f(\alpha_*).\end{aligned}$$

Thus, we have

$$\begin{aligned}&\mathbb{E}[n(f(\alpha_{t+1}) - f(\alpha_t) - \frac{L_{i_t}}{2} \|\alpha_{t+1,i_t} - \alpha_{t,i_t}\|_2^2) + f(\alpha_t) - f(\alpha_*)] \\ &\leq \frac{n}{2\eta_t} \mathbb{E}\left[\sum_{i=1}^n \|\alpha_{t,i} - \alpha_{*,i}\|_2^2 \right] - \frac{n}{2\eta_t} \mathbb{E}\left[\sum_{i=1}^n \|\alpha_{t+1,i} - \alpha_{*,i}\|_2^2 \right] \\ &- \frac{n}{2\eta_t} \mathbb{E}\left[\sum_{i=1}^n \|\alpha_{t,i} - \alpha_{t+1,i}\|_2^2 \right] + \sum_{i=1}^n \eta_t \sigma_i^2.\end{aligned}$$

Re-arranging this, we have

$$\begin{aligned}&\mathbb{E}[n(f(\alpha_{t+1}) - f(\alpha_*) - (n-1)(f(\alpha_t) - f(\alpha_*))] \\ &\leq \frac{n}{2\eta_t} \mathbb{E}\left[\sum_{i=1}^n \|\alpha_{t,i} - \alpha_{*,i}\|_2^2 \right] - \frac{n}{2\eta_t} \mathbb{E}\left[\sum_{i=1}^n \|\alpha_{t+1,i} - \alpha_{*,i}\|_2^2 \right] + \sum_{i=1}^n \eta_t \sigma_i^2 \\ &+ \mathbb{E}\left[\frac{n L_{i_t}}{2} \|\alpha_{t+1,i_t} - \alpha_{t,i_t}\|_2^2 \right] - \frac{n}{2\eta_t} \mathbb{E}\left[\sum_{i=1}^n \|\alpha_{t,i} - \alpha_{t+1,i}\|_2^2 \right] \\ &= \frac{n}{2\eta_t} \mathbb{E}\left[\sum_{i=1}^n \|\alpha_{t,i} - \alpha_{*,i}\|_2^2 \right] - \frac{n}{2\eta_t} \mathbb{E}\left[\sum_{i=1}^n \|\alpha_{t+1,i} - \alpha_{*,i}\|_2^2 \right] + \sum_{i=1}^n \eta_t \sigma_i^2 \\ &+ \mathbb{E}\left[\sum_{i=1}^n \frac{n L_i}{2} \|\alpha_{t+1,i} - \alpha_{t,i}\|_2^2 \right] - \frac{n}{2\eta_t} \mathbb{E}\left[\sum_{i=1}^n \|\alpha_{t,i} - \alpha_{t+1,i}\|_2^2 \right].\end{aligned}$$

If $\eta_t \leq \frac{1}{L}$, the sum of the last two terms is less than 0, then we have

$$\begin{aligned}&\mathbb{E}[f(\alpha_{t+1}) - f(\alpha_*)] \\ &\leq \mathbb{E}[(n-1)(f(\alpha_t) - f(\alpha_*)) - (n-1)(f(\alpha_{t+1}) - f(\alpha_*))] \\ &+ \frac{n}{2\eta_t} \mathbb{E}\left[\sum_{i=1}^n \|\alpha_{t,i} - \alpha_{*,i}\|_2^2 \right] - \frac{n}{2\eta_t} \mathbb{E}\left[\sum_{i=1}^n \|\alpha_{t+1,i} - \alpha_{*,i}\|_2^2 \right] + \sum_{i=1}^n \eta_t \sigma_i^2.\end{aligned}$$

Algorithm 5 SMD

Input: learning rate schedule $\{\eta_t\}_{t=1}^T$, starting point $\mathbf{w}_1$
for $t = 1, \ldots, T$ **do**
    Compute an unbiased gradient estimator $\mathbf{z}_t = \nabla g(\mathbf{w}_t; \zeta_t)$
    Update the model $\mathbf{w}$ by $\mathbf{w}_{t+1} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \mathbf{z}_t^\top(\mathbf{w} - \mathbf{w}_t) + \frac{1}{\eta_t} D_\varphi(\mathbf{w}, \mathbf{w}_t) + r(\mathbf{w})$.
end for

Averaging over $t = 1, \ldots, T$, we have

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T} f(\alpha_{t+1}) - f(\alpha_*)\right] \leq \frac{(n-1)(f(\alpha_1) - f(\alpha_*))}{T} + \frac{n}{2\eta T}\|\alpha_1 - \alpha_*\|_2^2 + \sum_{i=1}^{n} \eta\sigma_i^2,$$

which concludes the proof.

□

3.4 Stochastic Mirror Descent

The SGD update (3.2) and the SPGD update (3.19) can be generalized using the Bregman divergence instead of the Euclidean distance. Let φ be an α-strongly convex function with respect to a general norm $\|\cdot\|$. Recall the definition of Bregman divergence $D_\varphi(\mathbf{w}, \mathbf{w}')$ in Definition 1.7 induced by φ. Due to the strong convexity of φ, we have,

$$D_\varphi(\mathbf{w}, \mathbf{w}') \geq \frac{\alpha}{2}\|\mathbf{w} - \mathbf{w}'\|^2. \tag{3.32}$$

The stochastic mirror descent (SMD) update applied to non-smooth convex optimization problem (3.1) is given by

$$\mathbf{w}_{t+1} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \mathcal{G}(\mathbf{w}_t; \zeta_t)^\top(\mathbf{w} - \mathbf{w}_t) + \frac{1}{\eta_t} D_\varphi(\mathbf{w}, \mathbf{w}_t). \tag{3.33}$$

The SMD update applied to composite optimization problem (3.18) is given by

$$\mathbf{w}_{t+1} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \nabla g(\mathbf{w}_t; \zeta_t)^\top(\mathbf{w} - \mathbf{w}_t) + \frac{1}{\eta_t} D_\varphi(\mathbf{w}, \mathbf{w}_t) + r(\mathbf{w}). \tag{3.34}$$

Examples

Example 3.7 (Euclidean distance)**.** *By choosing $\varphi(\cdot) = \frac{1}{2}\|\cdot\|_2^2$, which is 1-strongly convex with respect to $\|\cdot\|_2$, the Bregman divergence reduces to the Euclidean distance, and the above updates simplify to SGD or SPGD.*

Example 3.8 (KL Divergence)**.** *Let us consider another example, where $r(\mathbf{w}) = \mathbb{I}_{0-\infty}(\mathbf{w} \in \Delta)$ and $\Delta_d = \{\mathbf{w} \in \mathbb{R}^d : \mathbf{w} \geq 0,\ \sum_{i=1}^d [\mathbf{w}]_i = 1\}$.*
By choosing $\varphi(\mathbf{w}) = \sum_{i=1}^d [\mathbf{w}]_i \log[\mathbf{w}]_i$, which is 1-strongly convex w.r.t $\|\cdot\|_1$ (cf. Lemma 1.10), the Bregman divergence reduces to the KL-divergence:

$$D_\varphi(\mathbf{w}, \mathbf{u}) = \sum_{i=1}^d [\mathbf{w}]_i \log \frac{[\mathbf{w}]_i}{[\mathbf{u}]_i},$$

and the SMD update (3.34) simplifies to

$$[\mathbf{w}_{t+1}]_i = \frac{[\mathbf{w}_t]_i \exp(-\eta_t [\nabla g(\mathbf{w}_t; \xi_t)]_i)}{\sum_{j=1}^d [\mathbf{w}_t]_j \exp(-\eta_t [\nabla g(\mathbf{w}_t; \xi_t)]_j)},$$

which is also known as stochastic exponential gradient descent.

Convergence Analysis

The following lemma is similar to Lemma 1.7.

Lemma 3.8 *If $r(\cdot)$ is convex and φ is α-strongly convex w.r.t a norm $\|\cdot\|$, with*

$$\mathbf{z}_1 = \arg\min_{\mathbf{w}} \mathbf{w}^\top \mathbf{a} + r(\mathbf{w}) + \frac{1}{\eta} D_\varphi(\mathbf{w}, \mathbf{z}),$$
$$\mathbf{z}_2 = \arg\min_{\mathbf{w}} \mathbf{w}^\top \mathbf{b} + r(\mathbf{w}) + \frac{1}{\eta} D_\varphi(\mathbf{w}, \mathbf{z}),$$

we have $\|\mathbf{z}_1 - \mathbf{z}_2\| \leq \frac{\eta}{\alpha}\|\mathbf{a} - \mathbf{b}\|_*$.

Proof. By the optimality of $\mathbf{z}_1$ and $\mathbf{z}_2$ we have

$$\mathbf{u} := \frac{\nabla\varphi(\mathbf{z}) - \nabla\varphi(\mathbf{z}_1)}{\eta} - \mathbf{a} \in \partial r(\mathbf{z}_1)$$
$$\mathbf{v} := \frac{\nabla\varphi(\mathbf{z}) - \nabla\varphi(\mathbf{z}_2)}{\eta} - \mathbf{b} \in \partial r(\mathbf{z}_2).$$

Since $r(\mathbf{x})$ is convex, we have

$$r(\mathbf{z}_1) \geq r(\mathbf{z}_2) + \mathbf{v}^\top(\mathbf{z}_1 - \mathbf{z}_2)$$
$$r(\mathbf{z}_2) \geq r(\mathbf{z}_1) + \mathbf{u}^\top(\mathbf{z}_2 - \mathbf{z}_1).$$

Adding them together, we have

$$0 \le (\mathbf{u}-\mathbf{v})^\top(\mathbf{z}_1-\mathbf{z}_2) = \frac{1}{\eta}(\eta\mathbf{b}-\eta\mathbf{a}+\nabla\varphi(\mathbf{z}_2)-\nabla\varphi(\mathbf{z}_1))^\top(\mathbf{z}_1-\mathbf{z}_2),$$

which implies

$$\frac{1}{\eta}(\nabla\varphi(\mathbf{z}_1)-\nabla\varphi(\mathbf{z}_2))^\top(\mathbf{z}_1-\mathbf{z}_2) \le (\mathbf{b}-\mathbf{a})^\top(\mathbf{z}_1-\mathbf{z}_2) \le \|\mathbf{b}-\mathbf{a}\|_*\|\mathbf{z}_1-\mathbf{z}_2\|.$$

Since φ is α-strongly convex, similar to Lemma 1.6 (c) we have

$$(\nabla\varphi(\mathbf{z}_1)-\nabla\varphi(\mathbf{z}_2))^\top(\mathbf{z}_1-\mathbf{z}_2) \ge \alpha\|\mathbf{z}_1-\mathbf{z}_2\|^2.$$

Combining the above two inequalities, we have $\|\mathbf{z}_1-\mathbf{z}_2\| \le \frac{\eta}{\alpha}\|\mathbf{a}-\mathbf{b}\|_*$. □

Lemma 3.9 (Generalized Three-point Equality) *For any $\mathbf{w}, \mathbf{w}_t, \mathbf{w}_{t+1}$, we have*

$$(\nabla\varphi(\mathbf{w}_t)-\nabla\varphi(\mathbf{w}_{t+1}))^\top(\mathbf{w}_{t+1}-\mathbf{w}) = D_\varphi(\mathbf{w},\mathbf{w}_t) - D_\varphi(\mathbf{w},\mathbf{w}_{t+1}) - D_\varphi(\mathbf{w}_{t+1},\mathbf{w}_t).$$

Proof.

$$\begin{aligned}
&D_\varphi(\mathbf{w},\mathbf{w}_{t+1}) - D_\varphi(\mathbf{w},\mathbf{w}_t)\\
&= -\varphi(\mathbf{w}_{t+1}) - \nabla\varphi(\mathbf{w}_{t+1})^\top(\mathbf{w}-\mathbf{w}_{t+1}) + \varphi(\mathbf{w}_t) + \nabla\varphi(\mathbf{w}_t)^\top(\mathbf{w}-\mathbf{w}_t)\\
&= (\nabla\varphi(\mathbf{w}_{t+1})-\nabla\varphi(\mathbf{w}_t))^\top(\mathbf{w}_{t+1}-\mathbf{w}) - \varphi(\mathbf{w}_{t+1}) + \varphi(\mathbf{w}_t) + \nabla\varphi(\mathbf{w}_t)^\top(\mathbf{w}_{t+1}-\mathbf{w}_t)\\
&= (\nabla\varphi(\mathbf{w}_{t+1})-\nabla\varphi(\mathbf{w}_t))^\top(\mathbf{w}_{t+1}-\mathbf{w}) - D_\varphi(\mathbf{w}_{t+1},\mathbf{w}_t).
\end{aligned}$$

Rearranging this equality finishes the proof. □

The following lemma is similar to Lemma 3.6.

Lemma 3.10 *Consider the update*

$$\mathbf{w}_{t+1} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \mathbf{z}_t^\top(\mathbf{w}-\mathbf{w}_t) + \frac{1}{\eta_t}D_\varphi(\mathbf{w},\mathbf{w}_t) + r(\mathbf{w}). \tag{3.35}$$

If $D_r(\mathbf{w},\mathbf{w}') \ge \mu D_\varphi(\mathbf{w},\mathbf{w}')$, then for any $\mathbf{w}$ we have

$$\begin{aligned}
\mathbf{z}_t^\top(\mathbf{w}_{t+1}-\mathbf{w}) + r(\mathbf{w}_{t+1}) - r(\mathbf{w}) \le{}& \frac{1}{\eta_t}D_\varphi(\mathbf{w},\mathbf{w}_t) - (\frac{1}{\eta_t}+\mu)D_\varphi(\mathbf{w},\mathbf{w}_{t+1})\\
&- \frac{1}{\eta_t}D_\varphi(\mathbf{w}_{t+1},\mathbf{w}_t).
\end{aligned}$$

Proof. By the first-order optimality condition of (3.35), we have

$$(\mathbf{z}_t + \partial r(\mathbf{w}_{t+1}) + \frac{1}{\eta_t}(\nabla\varphi(\mathbf{w}_{t+1})-\nabla\varphi(\mathbf{w}_t)))^\top(\mathbf{w}-\mathbf{w}_{t+1}) \ge 0. \tag{3.36}$$

By the assumption of r, we have

$$\mu D_\varphi(\mathbf{w}, \mathbf{w}_{t+1}) \leq r(\mathbf{w}) - r(\mathbf{w}_{t+1}) - \partial r(\mathbf{w}_{t+1})^\top (\mathbf{w} - \mathbf{w}_{t+1}).$$

Adding the above two inequalities, we have

$$\begin{aligned}
&\mathbf{z}_t^\top (\mathbf{w}_{t+1} - \mathbf{w}) + r(\mathbf{w}_{t+1}) - r(\mathbf{w}) \\
&\leq \frac{1}{\eta_t} (\nabla \varphi(\mathbf{w}_t) - \nabla \varphi(\mathbf{w}_{t+1}))^\top (\mathbf{w}_{t+1} - \mathbf{w}) - \mu D_\varphi(\mathbf{w}, \mathbf{w}_{t+1}) \\
&= \frac{1}{\eta_t} D_\varphi(\mathbf{w}, \mathbf{w}_t) - (\frac{1}{\eta_t} + \mu) D_\varphi(\mathbf{w}, \mathbf{w}_{t+1}) - \frac{1}{\eta_t} D_\varphi(\mathbf{w}_{t+1}, \mathbf{w}_t).
\end{aligned}$$

where the last equality uses Lemma 3.9. □

3.4.1 Non-smooth Composite Problems

Let us first analyze SMD (3.34) for the composite problem (3.18) under a modified Assumption.

Assumption 3.8. *Suppose the following conditions hold:*

(i) g is convex and L-smooth with respect to the norm $\|\cdot\|$, and r is convex.
(ii) There exists $\sigma > 0$ such that $\mathbb{E}_\zeta[\nabla g(\mathbf{w}; \zeta)] = \nabla g(\mathbf{w})$ and $\mathbb{E}_\zeta[\|\nabla g(\mathbf{w}; \zeta) - \nabla g(\mathbf{w})\|_*^2] \leq \sigma^2$ for all $\mathbf{w}$.

Theorem 3.10 *Suppose Assumption 3.8 holds. Let $\eta_t = \eta \leq \alpha/L$ and $\bar{\mathbf{w}}_T = \frac{1}{T}\sum_{t=1}^T \mathbf{w}_{t+1}$. After T iterations of SMD update (3.34) for the composite problem (3.18), we have*

$$\mathbb{E}[F(\bar{\mathbf{w}}_T) - F(\mathbf{w}_*)] \leq \frac{D_\varphi(\mathbf{w}_1, \mathbf{w}_*)}{\eta T} + \frac{\eta \sigma^2}{\alpha}.$$

If $\eta = \min\left(\frac{\alpha}{L}, \frac{\sqrt{\alpha D_\varphi(\mathbf{w}_1, \mathbf{w}_*)}}{\sqrt{T}\sigma}\right)$, then

$$\mathbb{E}\left[F(\bar{\mathbf{w}}_T) - F(\mathbf{w}_*)\right] \leq \frac{2\sigma\sqrt{D_\varphi(\mathbf{w}_1, \mathbf{w}_*)}}{\sqrt{T\alpha}} + \frac{2L D_\varphi(\mathbf{w}_1, \mathbf{w}_*)}{T\alpha}.$$

Why it matters

The key difference of the above result of SMD from that of SPGD in Theorem 3.5 lies in the divergence measure and the variance bound that is measured in the dual norm. Let us consider $r(\mathbf{w}) = \mathbb{I}_{0-\infty}(\mathbf{w} \in \Delta_d)$. With the Euclidean setup, the convergence upper bound is dominated by $O(\frac{\sigma_2 \|\mathbf{w}_1 - \mathbf{w}_*\|_2}{\sqrt{T}})$, where $\sigma_2^2 \geq \mathbb{E}\|\nabla g(\mathbf{w}, \zeta) - \nabla g(\mathbf{w})\|_2^2$ for all $\mathbf{w}, \zeta$.

In contrast, with the stochastic exponential gradient descent update, the convergence upper bound is dominated by $O(\frac{\sigma_\infty\sqrt{D_\varphi(\mathbf{w}_1,\mathbf{w}_*)}}{\sqrt{T}})$, where $\sigma_\infty^2 \geq \mathbb{E}\|\nabla g(\mathbf{w},\zeta) - \nabla g(\mathbf{w})\|_\infty^2$ for all $\mathbf{w},\zeta$. If we set $[\mathbf{w}_1]_i = \frac{1}{n}$ for all i, then we get $D_\varphi(\mathbf{w}_1,\mathbf{w}_*) \leq \log d$ for all $\mathbf{w}_* \in \Delta_d$. In addition, $\|\mathbf{w}_1 - \mathbf{w}_*\|_2$ could be $O(1)$. However, the constant σ_∞^2 can be smaller than σ_2^2 by a factor of d. Hence $\frac{\sigma_\infty\sqrt{D_\varphi(\mathbf{w}_1,\mathbf{w}_*)}}{\sigma_2\|\mathbf{w}_1-\mathbf{w}_*\|_2} = O(\frac{\log d}{\sqrt{d}})$, which indicates that stochastic exponential gradient descent may converge faster than SGD.

Proof. From Lemma 3.10, we have

$$\nabla g(\mathbf{w}_t,\zeta_t)^\top(\mathbf{w}_{t+1}-\mathbf{w}) + r(\mathbf{w}_{t+1}) - r(\mathbf{w}) \leq \frac{1}{\eta_t}(D_\varphi(\mathbf{w},\mathbf{w}_t) - D_\varphi(\mathbf{w},\mathbf{w}_{t+1})) - \frac{1}{\eta_t}D_\varphi(\mathbf{w}_{t+1},\mathbf{w}_t).$$

Same as (3.7) we have

$$g(\mathbf{w}_{t+1}) \leq g(\mathbf{w}) + \nabla g(\mathbf{w}_t)^\top(\mathbf{w}_t-\mathbf{w}) + \nabla g(\mathbf{w}_t)^\top(\mathbf{w}_{t+1}-\mathbf{w}_t) + \frac{L}{2}\|\mathbf{w}_{t+1}-\mathbf{w}_t\|^2.$$

Adding the above two inequalities for $\mathbf{w} = \mathbf{w}_*$, we have

$$F(\mathbf{w}_{t+1}) - F(\mathbf{w}_*) \leq \frac{1}{\eta_t}(D_\varphi(\mathbf{w},\mathbf{w}_t) - D_\varphi(\mathbf{w},\mathbf{w}_{t+1})) - \frac{1}{\eta_t}D_\varphi(\mathbf{w}_{t+1},\mathbf{w}_t) + \frac{L}{2}\|\mathbf{w}_{t+1}-\mathbf{w}_t\|^2 + (\nabla g(\mathbf{w}_t) - g(\mathbf{w}_t,\zeta_t))^\top(\mathbf{w}_{t+1}-\mathbf{w}_*). \tag{3.37}$$

Similar to the analysis of SPGD, we define:

$$\hat{\mathbf{w}}_{t+1} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \nabla g(\mathbf{w}_t)^\top(\mathbf{w}-\mathbf{w}_t) + \frac{1}{\eta_t}D_\varphi(\mathbf{w},\mathbf{w}_t) + r(\mathbf{w}),$$

which uses the full gradient $\nabla g(\mathbf{w}_t)$, making it independent of ζ_t. Then we have

$$\begin{aligned}&(\nabla g(\mathbf{w}_t) - g(\mathbf{w}_t,\zeta_t))^\top(\mathbf{w}_{t+1}-\mathbf{w}_*) \\ &\leq (\nabla g(\mathbf{w}_t) - g(\mathbf{w}_t,\zeta_t))^\top(\mathbf{w}_{t+1}-\hat{\mathbf{w}}_{t+1}) + (\nabla g(\mathbf{w}_t) - g(\mathbf{w}_t,\zeta_t))^\top(\hat{\mathbf{w}}_{t+1}-\mathbf{w}_*).\end{aligned} \tag{3.38}$$

In addition,

$$\begin{aligned}&(\nabla g(\mathbf{w}_t) - g(\mathbf{w}_t,\zeta_t))^\top(\mathbf{w}_{t+1}-\hat{\mathbf{w}}_{t+1}) \leq \|\nabla g(\mathbf{w}_t) - g(\mathbf{w}_t,\zeta_t)\|_*\|\mathbf{w}_{t+1}-\hat{\mathbf{w}}_{t+1}\| \\ &\leq \frac{\eta_t}{\alpha}\|\nabla g(\mathbf{w}_t) - g(\mathbf{w}_t,\zeta_t)\|_*^2,\end{aligned} \tag{3.39}$$

where the last inequality follows Lemma 3.8. Adding (3.37), (3.38) and (3.39) and using (3.32), we have

$$F(\mathbf{w}_{t+1}) - F(\mathbf{w}_*) \le \frac{1}{2\eta_t}(D_\varphi(\mathbf{w}, \mathbf{w}_t) - D_\varphi(\mathbf{w}, \mathbf{w}_{t+1})) - \left(\frac{\alpha}{2\eta_t} - \frac{L}{2}\right)\|\mathbf{w}_t - \mathbf{w}_{t+1}\|^2$$
$$+ (\nabla g(\mathbf{w}_t) - g(\mathbf{w}_t, \zeta_t))^\top (\mathbf{w}_{t+1} - \hat{\mathbf{w}}_{t+1}) + \frac{\eta_t}{\alpha}\|\nabla g(\mathbf{w}_t) - g(\mathbf{w}_t, \zeta_t)\|_*^2.$$

Taking expectation over ζ_t on both sides, we have

$$\mathbb{E}_{\zeta_t}[F(\mathbf{w}_{t+1}) - F(\mathbf{w}_*)]$$
$$\le \mathbb{E}_{\zeta_t}\left[\frac{1}{\eta_t}(D_\varphi(\mathbf{w}, \mathbf{w}_t) - D_\varphi(\mathbf{w}, \mathbf{w}_{t+1})) - \left(\frac{\alpha}{2\eta_t} - \frac{L}{2}\right)\|\mathbf{w}_t - \mathbf{w}_{t+1}\|^2\right] + \frac{\eta_t}{\alpha}\sigma^2.$$

If $\eta_t \le \frac{\alpha}{L}$, we have

$$\mathbb{E}_{\zeta_t}[F(\mathbf{w}_{t+1}) - F(\mathbf{w}_*)] \le \mathbb{E}_{\zeta_t}\left[\frac{1}{\eta_t}(D_\varphi(\mathbf{w}, \mathbf{w}_t) - D_\varphi(\mathbf{w}, \mathbf{w}_{t+1}))\right] + \frac{\eta_t}{\alpha}\sigma^2.$$

Summing over $t = 1, \ldots, T$, we have

$$\mathbb{E}\left[\frac{1}{\sum_{t=1}^T \eta_t}\sum_{t=1}^T \eta_t (F(\mathbf{w}_{t+1}) - F(\mathbf{w}_*))\right] \le \frac{D_\varphi(\mathbf{w}_1, \mathbf{w}_*)}{\sum_{t=1}^T \eta_t} + \frac{\sum_{t=1}^T \eta_t \sigma^2}{\alpha \sum_{t=1}^T \eta_t}.$$

Let $\eta_t = \eta$ and optimizing the upper bound over η finishes the proof. □

3.4.2 Non-smooth Problems

Next, we present the convergence analysis of SMD (3.33) for non-smooth convex objectives under the following assumption.

Assumption 3.9. *For any* $\mathbf{w}$*, we have* $\mathbb{E}_\zeta[\mathcal{G}(\mathbf{w}; \zeta)] \in \partial g(\mathbf{w})$ *and* $\mathbb{E}[\|\mathcal{G}(\mathbf{w}; \zeta)\|_*^2] \le G^2$.

Theorem 3.11 *Suppose Assumption 3.9 holds. Let the learning rate* $\{\eta_t\}$ *be* $\eta_t = \eta$ *and* $\bar{\mathbf{w}}_T = \frac{1}{T}\sum_{t=1}^T \mathbf{w}_t$*. After* T *iterations of SMD update (3.34), we have*

$$\mathbb{E}[g(\bar{\mathbf{w}}_T) - g(\mathbf{w}_*)] \le \frac{D_\varphi(\mathbf{w}_*, \mathbf{w}_1)}{\eta T} + \frac{\eta G^2}{2\alpha}.$$

If $\eta = \frac{\sqrt{2\alpha D_\varphi(\mathbf{w}_*, \mathbf{w}_1)}}{\sqrt{T}G}$*, then*

$$\mathbb{E}[g(\bar{\mathbf{w}}_T) - g(\mathbf{w}_*)] \le \frac{G\sqrt{2D_\varphi(\mathbf{w}_*, \mathbf{w}_1)}}{\sqrt{\alpha T}}.$$

Proof. From Lemma 3.10, we have

$$\mathcal{G}(\mathbf{w}_t, \zeta_t)^\top (\mathbf{w}_{t+1} - \mathbf{w}) \le \frac{1}{\eta_t}(D_\varphi(\mathbf{w}, \mathbf{w}_t) - D_\varphi(\mathbf{w}, \mathbf{w}_{t+1})) - \frac{1}{\eta_t} D_\varphi(\mathbf{w}_{t+1}, \mathbf{w}_t).$$

Rearranging it, we get

$$\begin{aligned}
&\eta_t \mathcal{G}(\mathbf{w}_t; \zeta_t)^\top (\mathbf{w}_t - \mathbf{w}) \\
&\quad \le D_\varphi(\mathbf{w}, \mathbf{w}_t) - D_\varphi(\mathbf{w}, \mathbf{w}_{t+1}) - D_\varphi(\mathbf{w}_{t+1}, \mathbf{w}_t) + \eta_t \mathcal{G}(\mathbf{w}_t; \zeta_t)^\top (\mathbf{w}_t - \mathbf{w}_{t+1}) \\
&\quad \le D_\varphi(\mathbf{w}, \mathbf{w}_t) - D_\varphi(\mathbf{w}, \mathbf{w}_{t+1}) - D_\varphi(\mathbf{w}_{t+1}, \mathbf{w}_t) \\
&\quad + \frac{\eta_t^2}{2\alpha} \|\mathcal{G}(\mathbf{w}_t; \zeta_t)\|_*^2 + \frac{\alpha}{2} \|\mathbf{w}_t - \mathbf{w}_{t+1}\|^2,
\end{aligned}$$

where the last inequality uses the Cauchy-Schwarz inequality. Using (3.32), we have

$$\eta_t \mathcal{G}(\mathbf{w}_t; \zeta_t)^\top (\mathbf{w}_t - \mathbf{w}) \le D_\varphi(\mathbf{w}, \mathbf{w}_t) - D_\varphi(\mathbf{w}, \mathbf{w}_{t+1}) + \frac{\eta_t^2}{2\alpha} \|\mathcal{G}(\mathbf{w}_t; \zeta_t)\|_*^2. \quad (3.40)$$

The remaining proof is similar to that of Theorem 3.2. □

3.5 Adaptive Gradient Method (AdaGrad)

The stochastic algorithms discussed so far are fairly general and were originally developed to address a wide range of problems, extending beyond those encountered specifically in machine learning. Nevertheless, the ERM problem of machine learning may exhibit some unique properties dependent on data. How to leverage them to develop a stochastic algorithm that could be potentially faster in practice?

Below, we introduce Adaptive Gradient Method (AdaGrad), which employs an adaptive step size, which incorporates knowledge of the geometry of the data observed in earlier iterations to perform more informative gradient-based learning.

While AdaGrad was considered an important breakthrough in machine learning, it indeed evolves from SMD. We use the same language as SMD to present AdaGrad and its analysis. Let us consider the smooth problem (3.1) and recall the update of SMD:

$$\mathbf{w}_{t+1} = \arg\min_{\mathbf{w} \in \mathbb{R}^d} \nabla g(\mathbf{w}_t; \zeta_t)^\top \mathbf{w} + \frac{1}{\eta} D_\varphi(\mathbf{w}, \mathbf{w}_t).$$

The key design to AdaGrad is to use a time-varying proximal function φ_t that changes across iterations. A specific way to construction φ_t is the following.

Let $H_t = \mathrm{diag}(s_{t,1}, \ldots, s_{t,d})$ be a diagonal positive matrix. Define $\varphi_t(\mathbf{w}) = \frac{1}{2}\mathbf{w}^\top H_t \mathbf{w}$ and a general norm $\|\mathbf{w}\|_H = \sqrt{\mathbf{w}^\top H \mathbf{w}}$. Then the Bregman divergence induced by φ_t becomes:

$$D_{\varphi_t}(\mathbf{w}, \mathbf{w}') = \frac{1}{2}(\mathbf{w} - \mathbf{w}')^\top H_t (\mathbf{w} - \mathbf{w}') = \frac{1}{2} \sum_{i=1}^d s_{t,i} (w_i - w_i')^2,$$

Algorithm 6 AdaGrad

Input: learning rate parameter η, starting point $\mathbf{w}_1$
for $t = 1, \ldots, T$ **do**
  Compute an unbiased gradient estimator $\mathbf{z}_t = \nabla g(\mathbf{w}_t; \zeta_t)$
  Update $s_{t,i} = \sqrt{\sum_{\tau=1}^{t} \|[\nabla g(\mathbf{w}_\tau; \zeta_t)]_i\|_2^2}, \forall i.$
  Update the model $\mathbf{w}$ by $\mathbf{w}_{t+1} = \mathbf{w}_t - \frac{\eta}{\mathbf{s}_t} \circ \mathbf{z}_t$
end for

which is 1-strongly convex w.r.t $\|\cdot\|_H$. The weights $\mathbf{s}_t = (s_{t,1}, \ldots, s_{t,d})$ are updated according to the following:

$$s_{t,i} = \sqrt{\sum_{\tau=1}^{t} [\nabla g(\mathbf{w}_\tau; \zeta_\tau)]_i^2}, \forall i, \tag{3.41}$$

which essentially measures the growth of stochastic gradients across all iterations before t.

Let $\mathbf{z}_t = \nabla g(\mathbf{w}_t, \zeta_t)$, and $\mathbf{m}_{1:t} = [\mathbf{z}_1, \ldots, \mathbf{z}_t]$, and $\mathbf{m}_{1:t,i}$ denotes its i-th row vector. Then $s_{t,i} = \|\mathbf{m}_{1:t,i}\|_2$. As a result, the updating step becomes

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta H_t^{-1} \nabla g(\mathbf{w}_t; \zeta_t) = \mathbf{w}_t - \frac{\eta}{\mathbf{s}_t} \circ \nabla g(\mathbf{w}_t; \zeta_t), \tag{3.42}$$

where $\circ$ denotes element-wise product. The full steps of AdaGrad are summarized in Algorithm 6.

Compared with SGD, there are two differences: (i) the effective step size $\frac{\eta}{\mathbf{s}_t}$ is adaptive that depends on the history of updates, hence depends on data sampled $\zeta_1, \ldots, \zeta_t$. This is the reason it is called adaptive step size; (ii) each coordinate of $\mathbf{w}$ will receive a different step size. This feature makes it useful to tackle deep neural networks as the parameters at each layer usually have different orders of gradient.

Convergence Analysis

Let the dual norm of $\|\cdot\|_H$ is given by $\|\mathbf{u}\|_{H^{-1}} = \sqrt{\mathbf{u}^\top H^{-1} \mathbf{u}}$. Then, φ_t is 1-strongly convex in terms of $\|\cdot\|_{H_t}$.

Lemma 3.11 *We have*

$$\sum_{t=1}^{T} \{D_{\varphi_t}(\mathbf{w}_*, \mathbf{w}_t) - D_{\varphi_t}(\mathbf{w}_*, \mathbf{w}_{t+1})\} \leq \frac{1}{2} \max_{t \leq T} \|\mathbf{w}_* - \mathbf{w}_t\|_\infty^2 \sum_{i=1}^{d} s_{T,i}.$$

Proof.

$$
\begin{aligned}
&\sum_{t=1}^{T}\{D_{\varphi_t}(\mathbf{w}_*,\mathbf{w}_t)-D_{\varphi_t}(\mathbf{w}_*,\mathbf{w}_{t+1})\}\\
&=\sum_{t=1}^{T}\{D_{\varphi_t}(\mathbf{w}_*,\mathbf{w}_t)-D_{\varphi_{t-1}}(\mathbf{w}_*,\mathbf{w}_t)+D_{\varphi_{t-1}}(\mathbf{w}_*,\mathbf{w}_t)-D_{\varphi_t}(\mathbf{w}_*,\mathbf{w}_{t+1})\}\\
&\le\sum_{t=1}^{T}\{D_{\varphi_t}(\mathbf{w}_*,\mathbf{w}_t)-D_{\varphi_{t-1}}(\mathbf{w}_*,\mathbf{w}_t)\}+D_{\varphi_0}(\mathbf{w}_*,\mathbf{w}_1)\\
&=D_{\varphi_0}(\mathbf{w}_*,\mathbf{w}_1)+\frac{1}{2}\sum_{t=1}^{T}(\mathbf{w}_*-\mathbf{w}_t)^\top(H_t-H_{t-1})(\mathbf{w}_*-\mathbf{w}_t).
\end{aligned}
$$

Since $\mathbf{s}_t \succeq \mathbf{s}_{t-1}$, we have

$$
\begin{aligned}
&\sum_{t=1}^{T}(\mathbf{w}_*-\mathbf{w}_t)^\top(H_t-H_{t-1})(\mathbf{w}_*-\mathbf{w}_t)=\sum_{t=1}^{T}\sum_{i=1}^{d}(s_{t,i}-s_{t-1,i})([\mathbf{w}_*]_i-[\mathbf{w}_t]_i)^2\\
&\le\max_{t\le T}\|\mathbf{w}_*-\mathbf{w}_t\|_\infty^2\sum_{t=1}^{T}\sum_{i=1}^{d}(s_{t,i}-s_{t-1,i})=\max_{t\le T}\|\mathbf{w}_*-\mathbf{w}_t\|_\infty^2\sum_{i=1}^{d}(s_{T,i}-s_{0,i}).
\end{aligned}
$$

Combining the above two inequalities, we have

$$
\begin{aligned}
&\sum_{t=1}^{T}D_{\varphi_t}(\mathbf{w}_*,\mathbf{w}_t)-D_{\varphi_t}(\mathbf{w}_*,\mathbf{w}_{t+1})\\
&\le D_{\varphi_0}(\mathbf{w}_*,\mathbf{w}_1)+\frac{1}{2}\max_{t\le T}\|\mathbf{w}_*-\mathbf{w}_{t+1}\|_\infty^2\sum_{i=1}^{d}(s_{T+1,i}-s_{1,i})\\
&\le\frac{1}{2}\|\mathbf{w}_1-\mathbf{w}_*\|_\infty^2\sum_{i=1}^{d}s_{0,i}+\frac{1}{2}\max_{t\le T}\|\mathbf{w}_*-\mathbf{w}_t\|_\infty^2\sum_{i=1}^{d}(s_{T,i}-s_{0,i})\\
&\le\frac{1}{2}\max_{t\le T}\|\mathbf{w}_*-\mathbf{w}_t\|_\infty^2\sum_{i=1}^{d}s_{T,i}.
\end{aligned}
$$

□

Lemma 3.12 *We have*

$$
\sum_{t=1}^{T}\|\nabla g(\mathbf{w}_t;\zeta_t)\|_{H_t^{-1}}^2\le 2\sum_{i=1}^{d}s_{T,i}.
$$

Proof. Let us first prove a general result in the following: for a general real-value sequence $\{a_t\}$, we have

$$\sum_{t=1}^{T} \frac{a_t^2}{\|a_{1:t}\|_2} \leq 2 \sum_{t=1}^{T} \|a_{1:T}\|_2, \tag{3.43}$$

where $a_{1:t} = (a_1, \ldots, a_t)$. We prove this by induction. First, it holds trivially for $t = 1$. Now, assume it holds for $T - 1$, we prove it holds for T.

$$\sum_{t=1}^{T} \frac{a_t^2}{\sqrt{\sum_{\tau=1}^{t} a_t^2}} = \sum_{t=1}^{T-1} \frac{a_t^2}{\sqrt{\sum_{\tau=1}^{t} a_t^2}} + \frac{a_T^2}{\|a_{1:T}\|_2} \leq 2 \sum_{t=1}^{T} \|a_{1:T-1}\|_2 + \frac{a_T^2}{\|a_{1:T}\|_2}.$$

Let $b_T = \sqrt{\sum_{t=1}^{T} a_t^2}$, then we have

$$2 \sum_{t=1}^{T} \|a_{1:T-1}\|_2 + \frac{a_T^2}{\|a_{1:T}\|_2} = 2\sqrt{b_T^2 - a_T^2} + \frac{a_T^2}{\sqrt{b_T^2}}.$$

Since $\sqrt{\cdot}$ is a concave function, applying $\sqrt{x + \delta} \leq \sqrt{x} + \delta \frac{1}{2\sqrt{x}}$ we have

$$\sqrt{b_T^2 - a_T^2} \leq \sqrt{b_T^2} - (a_T^2) \frac{1}{2\sqrt{b_T^2}}.$$

Hence, $2\sqrt{b_T^2 - a_T^2} + \frac{a_T^2}{\sqrt{b_T^2}} \leq 2\sqrt{b_T^2}$. Thus, we prove (3.43) for T.

Next, we apply this result to the following:

$$\begin{aligned}
&\sum_{t=1}^{T} \|\nabla g(\mathbf{w}_t; \zeta_t)\|_{H_t^{-1}}^2 = \sum_{t=1}^{T} \nabla g(\mathbf{w}_t; \zeta_t)^\top \mathrm{diag}(\mathbf{s}_t)^{-1} \nabla g(\mathbf{w}_t; \zeta_t) \\
&= \sum_{i=1}^{d} \frac{[\nabla g(\mathbf{w}_t; \zeta_t)]_i^2}{\sqrt{\sum_{\tau=1}^{t} [\nabla g(\mathbf{w}_\tau; \zeta_\tau)]_i^2}} \leq \sum_{i=1}^{d} 2\sqrt{\sum_{\tau=1}^{t} [\nabla g(\mathbf{w}_\tau; \zeta_\tau)]_i^2}.
\end{aligned}$$

□

Theorem 3.12 *Let* $\bar{\mathbf{w}}_T = \frac{1}{T} \sum_{t=1}^{T} \mathbf{w}_t$*, then AdaGrad guarantees that*

$$\begin{aligned}
\mathbb{E}[g(\bar{\mathbf{w}}_t) - g(\mathbf{w}_*)] \leq & \frac{\mathbb{E}\left[\max_{t \leq T} \|\mathbf{w}_* - \mathbf{w}_t\|_\infty^2 \sum_{i=1}^{d} \|\mathbf{m}_{1:T,i}\|_2\right]}{2\eta T} \\
& + \frac{\eta \mathbb{E}\left[\sum_{i=1}^{d} \|\mathbf{m}_{1:T,i}\|_2\right]}{T}.
\end{aligned}$$

If $\max_t \|\mathbf{w}_* - \mathbf{w}_t\|_\infty \leq D_\infty$ *and* $\eta = D_\infty / \sqrt{2}$*, we have*

$$\mathbb{E}[g(\bar{\mathbf{w}}_t) - g(\mathbf{w}_*)] \leq \frac{\sqrt{2} D_\infty \mathbb{E}\left[\sum_{i=1}^{d} \|\mathbf{m}_{1:T,i}\|_2\right]}{T}.$$

Why it matters

The above result shows the convergence rate depends on the growth rate of the cumulative stochastic gradient $\sum_{i=1}^{d} \|\mathbf{m}_{1:T,i}\|_2$. In the worst case, it grows at a rate of $O(\sqrt{T})$, inducing a convergence rate of $O(1/\sqrt{T})$, similar to SGD. However, when the cumulative stochastic gradient grows slower than $O(\sqrt{T})$, AdaGrad will enjoy a convergence rate of $o(1/\sqrt{T})$.

Let us consider the following linear model with sparse random data scenario, where $g(\mathbf{w}_t, \zeta_t) = [1 - \mathbf{w}_t^\top \zeta_t]_+$ and the data vectors $\zeta_t \in \{-1, 0, 1\}^d$. Assume that at in each round t, feature i appears with probability $p_i = \min\{1, ci^{-\alpha}\}$ for some $\alpha \in (1, \infty)$ and a dimension-independent constant c. Then we have

$$\mathbb{E}\left[\sum_{i=1}^{d} \|\mathbf{m}_{1:T,i}\|_2\right] = \mathbb{E}\left[\sum_{i=1}^{d} \sqrt{\left|t : \mathbf{z}_{t,i} = 1\right|}\right] \leq \sum_{i=1}^{d} \sqrt{\mathbb{E}\left[\left|t : \mathbf{z}_{t,i} = 1\right|\right]} = \sum_{i=1}^{d} \sqrt{p_i T}.$$

by Jensen's inequality. In the rightmost sum, we have $c\sum_{i=1}^{d} i^{-\alpha/2} = O(\log d)$ for $\alpha \geq 2$, and $\sum_{i=1}^{d} i^{-\alpha/2} = O(d^{1-\alpha/2})$ for $\alpha \in (1, 2)$. If $\mathbf{w}_t$ is restricted in a domain $\mathcal{W} = \{\mathbf{w} : \|\mathbf{w}\|_\infty \leq 1\}$, then $D_\infty = 2$, and the convergence rate of AdaGrad becomes $O(\max\{\log d, d^{1-\alpha/2}\}/\sqrt{T})$. For contrast, the convergence rate of SGD in Theorem 3.2 is $O(\sqrt{d/T})$. So we see that in this sparse yet heavy tailed feature setting, AdaGrad's convergence bound can be exponentially smaller in the dimension d than the non-adaptive bound of SGD.

Proof. Similar to (3.40) in the proof of Theorem 3.11, we have

$$\eta\langle \nabla g(\mathbf{w}_t; \zeta_t), \mathbf{w}_t - \mathbf{w}\rangle \leq D_{\varphi_t}(\mathbf{w}, \mathbf{w}_t) - D_{\varphi_t}(\mathbf{w}, \mathbf{w}_{t+1}) + \frac{\eta_t^2}{2}\|\nabla g(\mathbf{w}_t; \zeta_t)\|_{H_t^{-1}}^2. \quad (3.44)$$

Taking expectation and summation over $t = 1, \ldots, T$, we have

$$\sum_{t=1}^{T} \eta\mathbb{E}[g(\mathbf{w}_t) - g(\mathbf{w}_*)] \leq \mathbb{E}\left[\sum_{t=1}^{T} D_{\varphi_t}(\mathbf{w}, \mathbf{w}_t) - D_{\varphi_t}(\mathbf{w}, \mathbf{w}_{t+1})\right] + \mathbb{E}\left[\sum_{t=1}^{T} \frac{\eta^2}{2}\|\nabla g(\mathbf{w}_t; \zeta_t)\|_{H_t^{-1}}^2\right].$$

Using the results from the two lemmas above, we conclude the proof.

□

3.6 Stochastic Gradient Descent Ascent

In this section, we consider stochastic convex–concave min–max optimization problems:

$$\min_{\mathbf{w}\in\mathcal{W}} \max_{\mathbf{u}\in\mathcal{U}} f(\mathbf{w},\mathbf{u}) := \mathbb{E}_{\zeta}\big[f(\mathbf{w},\mathbf{u};\zeta)\big].$$

This class of problems has two important applications in machine learning: (1) it serves as a foundation for directly formulating learning tasks (e.g., the DRO problem (2.11)); (2) it provides a tool for reformulating standard minimization problems to enable more efficient optimization.

A solution of interest is the so-called saddle point $(\mathbf{w}_*,\mathbf{u}_*)\in\mathcal{W}\times\mathcal{U}$ satisfying:

$$f(\mathbf{w}_*,\mathbf{u}) \le f(\mathbf{w}_*,\mathbf{u}_*) \le f(\mathbf{w},\mathbf{u}_*), \forall \mathbf{w}\in\mathcal{W}, \mathbf{u}\in\mathcal{U}.$$

In many machine learning applications, we may be only interested in finding a global optimal solution to the objective $F(\mathbf{w}) = \max_{\mathbf{u}\in\mathcal{U}} f(\mathbf{w},\mathbf{u})$. It is easy to see that if $(\mathbf{w}_*,\mathbf{u}_*)$ is a saddle point, then $\mathbf{w}_*$ is a global optimal solution to $F(\mathbf{w})$. This can be seen from

$$\max_{\mathbf{u}\in\mathcal{U}} f(\mathbf{w}_*,\mathbf{u}) \le f(\mathbf{w}_*,\mathbf{u}_*) \le f(\mathbf{w},\mathbf{u}_*) \le \max_{\mathbf{u}\in\mathcal{U}} f(\mathbf{w},\mathbf{u}).$$

For a point $(\mathbf{w},\mathbf{u})\in\mathcal{W}\times\mathcal{U}$, a convergence measure is defined by the duality gap:

$$\Delta(\mathbf{w},\mathbf{u}) = \max_{\mathbf{u}'\in\mathcal{U}} f(\mathbf{w},\mathbf{u}') - \min_{\mathbf{w}'\in\mathcal{W}} f(\mathbf{w}',\mathbf{u}).$$

A simple method for solving the convex-concave min-max problem is the stochastic gradient descent ascent (SGDA) algorithm, which is an extension of SGD. It employs two key updates:

$$\begin{aligned}
\mathbf{w}_{t+1} &= \arg\min_{\mathbf{w}\in\mathcal{W}} \partial_1 f(\mathbf{w}_t,\mathbf{u}_t;\zeta_t)^\top(\mathbf{w}-\mathbf{w}_t) + \frac{1}{2\eta_1}\|\mathbf{w}-\mathbf{w}_t\|_2^2 \\
\mathbf{u}_{t+1} &= \arg\min_{\mathbf{u}\in\mathcal{U}} -\partial_2 f(\mathbf{w}_t,\mathbf{u}_t;\zeta_t)^\top(\mathbf{u}-\mathbf{u}_t) + \frac{1}{2\eta_2}\|\mathbf{u}-\mathbf{u}_t\|_2^2,
\end{aligned} \tag{3.45}$$

where $\partial_1 f(\mathbf{w},\mathbf{u};\zeta)$ and $\partial_2 f(\mathbf{w},\mathbf{u};\zeta)$ denote the stochastic partial subgradients such that $\mathbb{E}_\zeta[\partial_1 f(\mathbf{w},\mathbf{u};\zeta)]\in\partial_1 f(\mathbf{w},\mathbf{u})$ and $\mathbb{E}_\zeta[\partial_2 f(\mathbf{w},\mathbf{u};\zeta)]\in\partial_2 f(\mathbf{w},\mathbf{u})$.

Convergence Analysis

Below, we analyze the convergence rate of SGDA under the following assumptions.

Assumption 3.10. *Suppose the following conditions hold:*

(i) $f(\mathbf{w},\mathbf{u})$ *is convex w.r.t* $\mathbf{w}$ *and concave w.r.t* $\mathbf{u}$.

Algorithm 7 SGDA

Input: learning rates $\{\eta_1, \eta_2\}$, starting points $\mathbf{w}_1, \mathbf{u}_1$
for $t = 1, \ldots, T$ **do**
  Compute unbiased gradient estimators $\mathbf{z}_{1,t} = \partial_1 f(\mathbf{w}_t; \zeta_t)$ and $\mathbf{z}_{2,t} = \partial_2 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t)$
  Update the primal variable $\mathbf{w}$ by $\mathbf{w}_{t+1} = \arg\min_{\mathbf{w}\in\mathcal{W}} \mathbf{z}_{1,t}^\top (\mathbf{w} - \mathbf{w}_t) + \frac{1}{2\eta_1}\|\mathbf{w} - \mathbf{w}_t\|_2^2$.
  Update the dual variable $\mathbf{u}$ by $\mathbf{u}_{t+1} = \arg\min_{\mathbf{u}\in\mathcal{U}} -\mathbf{z}_{2,t}^\top (\mathbf{u} - \mathbf{u}_t) + \frac{1}{2\eta_2}\|\mathbf{u} - \mathbf{u}_t\|_2^2$.
end for

(ii) There exist $G_1, G_2 > 0$ *such that*

$$\mathbb{E}_\zeta[\|\partial_1 f(\mathbf{w}, \mathbf{u}; \zeta)\|_2^2] \le G_1^2, \forall \mathbf{w} \in \mathcal{W}, \mathbf{u} \in \mathcal{U}, \tag{3.46}$$

$$\mathbb{E}_\zeta[\|\partial_2 f(\mathbf{w}, \mathbf{u}; \zeta)\|_2^2] \le G_2^2, \forall \mathbf{w} \in \mathcal{W}, \mathbf{u} \in \mathcal{U}. \tag{3.47}$$

(iii) $\max_{\mathbf{w}\in\mathcal{W}, \mathbf{w}'\in\mathcal{W}} \|\mathbf{w} - \mathbf{w}'\| \le D_1$ *and* $\max_{\mathbf{u}\in\mathcal{U}, \mathbf{u}'\in\mathcal{U}} \|\mathbf{u} - \mathbf{u}'\| \le D_2$.

Lemma 3.13 *Let us consider a martingale difference sequence* $\{\delta_t\}_{t\ge1}$ *and a sequence* $\{y_t\}_{t\ge1}$*:*

$$y_{t+1} = \arg\min_{v\in\mathcal{V}} \{-\delta_t^\top v + \alpha D_\psi(v, y_t)\}.$$

If ψ *is* μ_ψ*-strongly convex w.r.t.* $\|\cdot\|$ *(*$\mu_\psi > 0$*). For any* v *(that possibly depends on* $\{\delta_t\}$*) we have*

$$\mathbb{E}\left[\delta_t^\top v\right] \le \mathbb{E}\left[\alpha D_\psi(v, y_t) - \alpha D_\psi(v, y_{t+1}) + \frac{1}{2\alpha\mu_\psi}\|\delta_t\|_*^2\right].$$

Why it matters

In standard minimization problems, the convergence measure is usually defined with respect to the optimal solution $\mathbf{w}_*$, which is fixed and independent of the randomness introduced by the algorithm. In contrast, in stochastic min–max optimization we are concerned with the duality gap $\Delta(\mathbf{w}, \mathbf{u}) = \max_{\mathbf{u}'\in\mathcal{U}} f(\mathbf{w}, \mathbf{u}') - \min_{\mathbf{w}'\in\mathcal{W}} f(\mathbf{w}', \mathbf{u})$, where the optimal $\mathbf{w}'$ and $\mathbf{u}'$ depend on the current random iterates $(\mathbf{w}, \mathbf{u})$. This dependency introduces additional subtleties into the analysis.

The preceding lemma applies to any random variable v that may depend on the entire randomness of the algorithm, and will be useful for our analysis. Recall that a sequence $\{X_t\}$ is a *martingale difference sequence* if the conditional expectation of each variable given the past is zero, i.e., $\mathbb{E}[X_t \mid X_1, \ldots, X_{t-1}] = 0$.

Proof. Applying Lemma 3.10 to the update of y_{t+1}, we have

$$\mathbb{E}\left[-\delta_t^\top (y_{t+1} - v)\right] \le \mathbb{E}\left[\alpha D_\psi(y, y_t) - \alpha D_\psi(y, y_{t+1}) - \alpha D_\psi(y_{t+1}, y_t)\right].$$

Hence,

$$\begin{aligned}\mathbb{E}\left[\delta_t^\top (v - y_t)\right] &\leq \mathbb{E}\left[\alpha D_\psi(v, y_t) - \alpha D_\psi(v, y_{t+1}) - \alpha D_\psi(y_{t+1}, y_t)\right] \\ &\quad + \mathbb{E}[\delta_t^\top (y_{t+1} - y_t)] \\ &\leq \mathbb{E}\left[\alpha D_\psi(v, y_t) - \alpha D_\psi(v, y_{t+1})\right] \\ &- \mathbb{E}\left[\frac{\alpha\mu_\psi}{2}\|y_{t+1} - y_t\|^2 + \frac{\mu_\psi\alpha}{2}\|y_{t+1} - y_t\|^2 + \frac{1}{2\mu_\psi\alpha}\|\delta_t\|_*^2\right].\end{aligned}$$

Since $\mathbb{E}[\delta_t] = 0$ and y_t is independent of δ_t, we have $\mathbb{E}[\delta_t^\top y_t] = 0$. As a result,

$$\mathbb{E}[\delta_t^\top v] \leq \mathbb{E}\left[\alpha D_\psi(v, y_t) - \alpha D_\psi(v, y_{t+1})\right] + \frac{1}{2\mu_\psi\alpha}\mathbb{E}\left[\|\delta_t\|_*^2\right].$$

□

Theorem 3.13 *Let* $\bar{\mathbf{w}}_T = \frac{1}{T}\sum_{t=1}^T \mathbf{w}_t, \bar{\mathbf{u}}_T = \frac{1}{T}\sum_{t=1}^T \mathbf{u}_t$. *After T iterations, SGDA (3.45) guarantees that*

$$\mathbb{E}[\Delta(\bar{\mathbf{w}}_T, \bar{\mathbf{u}}_T)] \leq \frac{D_1^2}{\eta_1 T} + \frac{D_2^2}{\eta_2 T} + \frac{5\eta_1 G_1^2}{2} + \frac{5\eta_2 G_2^2}{2}.$$

If we set $\eta_1 = O(\frac{D_1}{G_1\sqrt{T}})$ *and* $\eta_2 = O(\frac{D_2}{G_2\sqrt{T}})$, *we have*

$$\mathbb{E}[\Delta(\bar{\mathbf{w}}_T, \bar{\mathbf{u}}_T)] \leq O\left(\frac{D_1 G_1}{\sqrt{T}} + \frac{D_2 G_2}{\sqrt{T}}\right).$$

Proof. Similar to (3.10), for the primal update and dual update for any $\mathbf{w} \in \mathcal{W}, \mathbf{u} \in \mathcal{U}$ we have

$$\begin{aligned}&\partial_1 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t)^\top (\mathbf{w}_t - \mathbf{w}) \leq \\ &\qquad \frac{1}{2\eta_1}\|\mathbf{w}_t - \mathbf{w}\|_2^2 - \frac{1}{2\eta_1}\|\mathbf{w}_{t+1} - \mathbf{w}\|_2^2 + \frac{1}{2}\eta_1\|\partial_1 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t)\|_2^2 \\ &-\partial_2 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t)^\top (\mathbf{u}_t - \mathbf{u}) \leq \\ &\qquad \frac{1}{2\eta_2}\|\mathbf{u}_t - \mathbf{u}\|_2^2 - \frac{1}{2\eta_2}\|\mathbf{u}_{t+1} - \mathbf{u}\|_2^2 + \frac{1}{2}\eta_2\|\partial_2 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t)\|_2^2.\end{aligned}$$

The difference from the SGD analysis is that we cannot fix $\mathbf{w}$ as $\mathbf{w}_*$ and fix $\mathbf{u}$ as $\mathbf{u}_*$, which will not yield the duality gap measure. Indeed, at the end we need to take max over $\mathbf{w} \in \mathcal{W}$ and min over $\mathbf{u} \in \mathcal{U}$ to obtain the duality gap, making them dependent on the randomness.

To proceed, we have

$$
\begin{aligned}
&\partial_1 f(\mathbf{w}_t, \mathbf{u}_t)^\top (\mathbf{w}_t - \mathbf{w}) \leq \frac{1}{2\eta_1}\|\mathbf{w}_t - \mathbf{w}\|_2^2 - \frac{1}{2\eta_1}\|\mathbf{w}_{t+1} - \mathbf{w}\|_2^2 \\
&+ \frac{1}{2}\eta_1 \|\partial_1 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t)\|_2^2 + (\partial_1 f(\mathbf{w}_t, \mathbf{u}_t) - \partial_1 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t))^\top (\mathbf{w}_t - \mathbf{w}) \\
&- \partial_2 f(\mathbf{w}_t, \mathbf{u}_t)^\top (\mathbf{u}_t - \mathbf{u}) \leq \frac{1}{2\eta_2}\|\mathbf{u}_t - \mathbf{u}\|_2^2 - \frac{1}{2\eta_2}\|\mathbf{u}_{t+1} - \mathbf{u}\|_2^2 \\
&+ \frac{1}{2}\eta_2 \|\partial_2 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t)\|_2^2 + (\partial_2 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t) - \partial_2 f(\mathbf{w}_t, \mathbf{u}_t))^\top (\mathbf{u}_t - \mathbf{u}).
\end{aligned}
$$

Adding these inequalities we have

$$
\begin{aligned}
&\partial_1 f(\mathbf{w}_t, \mathbf{u}_t)^\top (\mathbf{w}_t - \mathbf{w}) - \partial_2 f(\mathbf{w}_t, \mathbf{u}_t)^\top (\mathbf{u}_t - \mathbf{u}) \\
&\leq \frac{1}{2\eta_1}\left(\|\mathbf{w}_t - \mathbf{w}\|_2^2 - \|\mathbf{w}_{t+1} - \mathbf{w}\|_2^2\right) + \frac{1}{2\eta_2}\left(\|\mathbf{u}_t - \mathbf{u}\|_2^2 - \|\mathbf{u}_{t+1} - \mathbf{u}\|_2^2\right) \\
&+ \frac{1}{2}\eta_1 \|\partial_1 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t)\|_2^2 + \frac{1}{2}\eta_2 \|\partial_2 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t)\|_2^2 \\
&+ (\partial_1 f(\mathbf{w}_t, \mathbf{u}_t) - \partial_1 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t))^\top (\mathbf{w}_t - \mathbf{w}) \\
&+ (\partial_2 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t) - \partial_2 f(\mathbf{w}_t, \mathbf{u}_t))^\top (\mathbf{u}_t - \mathbf{u}).
\end{aligned}
$$

Due to the convexity and concavity of $f(\mathbf{w}, \mathbf{u})$ in terms of $\mathbf{w}, \mathbf{u}$, respectively, we have

$$
\begin{aligned}
f(\mathbf{w}_t, \mathbf{u}_t) - f(\mathbf{w}, \mathbf{u}_t) &\leq \partial_1 f(\mathbf{w}_t, \mathbf{u}_t)^\top (\mathbf{w}_t - \mathbf{w}), \\
f(\mathbf{w}_t, \mathbf{u}) - f(\mathbf{w}_t, \mathbf{u}_t) &\leq -\partial_2 f(\mathbf{w}_t, \mathbf{u}_t)^\top (\mathbf{u}_t - \mathbf{u}).
\end{aligned}
$$

Adding these two equalities, we have

$$
f(\mathbf{w}_t, \mathbf{u}) - f(\mathbf{w}, \mathbf{u}_t) \leq \partial_1 f(\mathbf{w}_t, \mathbf{u}_t)^\top (\mathbf{w}_t - \mathbf{w}) - \partial_2 f(\mathbf{w}_t, \mathbf{u}_t)^\top (\mathbf{u}_t - \mathbf{u}).
$$

As a result, we have

$$
\begin{aligned}
&f(\mathbf{w}_t, \mathbf{u}) - f(\mathbf{w}, \mathbf{u}_t) \\
&\leq \frac{1}{2\eta_1}\left(\|\mathbf{w}_t - \mathbf{w}\|_2^2 - \|\mathbf{w}_{t+1} - \mathbf{w}\|_2^2\right) + \frac{1}{2\eta_2}\left(\|\mathbf{u}_t - \mathbf{u}\|_2^2 - \|\mathbf{u}_{t+1} - \mathbf{u}\|_2^2\right) \\
&+ \frac{1}{2}\eta_1 \|\partial_1 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t)\|_2^2 + \frac{1}{2}\eta_2 \|\partial_2 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t)\|_2^2 \\
&+ (\partial_1 f(\mathbf{w}_t, \mathbf{u}_t) - \partial_1 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t))^\top (\mathbf{w}_t - \mathbf{w}) \\
&+ (\partial_2 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t) - \partial_2 f(\mathbf{w}_t, \mathbf{u}_t))^\top (\mathbf{u}_t - \mathbf{u}).
\end{aligned}
$$

Taking average over $t = 1, \ldots, T$, we have

$$
\begin{aligned}
f(\bar{\mathbf{w}}_T, \mathbf{u}) - f(\mathbf{w}, \bar{\mathbf{u}}_T) &\leq \frac{1}{T}\sum_{t=1}^{T}\left(f(\mathbf{w}_t, \mathbf{u}) - f(\mathbf{w}, \mathbf{u}_t)\right) \\
&\leq \frac{1}{2\eta_1 T}\|\mathbf{w}_1 - \mathbf{w}\|_2^2 + \frac{1}{2\eta_2 T}\|\mathbf{u}_1 - \mathbf{u}\|_2^2 \\
&+ \frac{\eta_1}{2T}\sum_{t=1}^{T}\|\partial_1 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t)\|_2^2 + \frac{\eta_2}{2T}\sum_{t=1}^{T}\|\partial_2 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t)\|_2^2 \\
&+ \frac{1}{T}\sum_{t=1}^{T}(\partial_1 f(\mathbf{w}_t, \mathbf{u}_t) - \partial_1 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t))^\top(\mathbf{w}_t - \mathbf{w}) \\
&+ \frac{1}{T}\sum_{t=1}^{T}(\partial_2 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t) - \partial_2 f(\mathbf{w}_t, \mathbf{u}_t))^\top(\mathbf{u}_t - \mathbf{u}).
\end{aligned}
$$

Let $\mathbf{w}, \mathbf{u}$ be the solution to $\max_{\mathbf{w}\in\mathcal{W}, \mathbf{u}\in\mathcal{U}} f(\bar{\mathbf{w}}_T, \mathbf{u}) - f(\mathbf{w}, \bar{\mathbf{u}}_T)$, which are random variables. Taking expectation over both sides, we have

$$
\begin{aligned}
\mathbb{E}[\Delta(\bar{\mathbf{w}}_T, \bar{\mathbf{u}}_T)] \leq & \frac{1}{2\eta_1 T}\|\mathbf{w}_1 - \mathbf{w}\|_2^2 + \frac{1}{2\eta_2 T}\|\mathbf{u}_1 - \mathbf{u}\|_2^2 + \frac{\eta_1 G_1^2}{2} + \frac{\eta_2 G_2^2}{2} \\
&+ \frac{1}{T}\mathbb{E}\left[\sum_{t=1}^{T}(\partial_1 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t) - \partial_1 f(\mathbf{w}_t, \mathbf{u}_t))^\top \mathbf{w}\right] \\
&+ \frac{1}{T}\mathbb{E}\left[\sum_{t=1}^{T}(\partial_2 f(\mathbf{w}_t, \mathbf{u}_t) - \partial_2 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t))^\top \mathbf{u}\right].
\end{aligned} \tag{3.48}
$$

Next, we apply Lemma 3.13 to bound the last two terms. To this end, we introduce two virtual sequences with $\hat{\mathbf{w}}_1 = \mathbf{w}_1, \hat{\mathbf{u}}_1 = \mathbf{u}_1$:

$$
\begin{aligned}
\hat{\mathbf{w}}_{t+1} &= \arg\min_{\mathbf{w}\in\mathcal{W}} -(\partial_1 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t) - \partial_1 f(\mathbf{w}_t, \mathbf{u}_t))^\top \mathbf{w} + \frac{1}{2\eta_1}\|\mathbf{w} - \hat{\mathbf{w}}_t\|_2^2 \\
\hat{\mathbf{u}}_{t+1} &= \arg\min_{\mathbf{u}\in\mathcal{U}} (\partial_2 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t) - \partial_2 f(\mathbf{w}_t, \mathbf{u}_t))^\top \mathbf{u} + \frac{1}{2\eta_2}\|\mathbf{u} - \hat{\mathbf{u}}_t\|_2^2.
\end{aligned}
$$

Applying Lemma 3.13, we have

$$
\begin{aligned}
&\mathbb{E}\left[(\partial_1 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t) - \partial_1 f(\mathbf{w}_t, \mathbf{u}_t))^\top \mathbf{w}\right] \leq \frac{1}{2\eta_1}\left(\|\hat{\mathbf{w}}_t - \mathbf{w}\|_2^2 - \|\hat{\mathbf{w}}_{t+1} - \mathbf{w}\|_2^2\right) \\
&+ \frac{\eta_1}{2}\mathbb{E}[\|\partial_1 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t) - \partial_1 f(\mathbf{w}_t, \mathbf{u}_t)\|_2^2] \\
&\mathbb{E}\left[(\partial_2 f(\mathbf{w}_t, \mathbf{u}_t) - \partial_2 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t))^\top \mathbf{u}\right] \leq \frac{1}{2\eta_2}\left(\|\hat{\mathbf{u}}_t - \mathbf{u}\|_2^2 - \|\hat{\mathbf{u}}_{t+1} - \mathbf{u}\|_2^2\right) \\
&+ \frac{\eta_2}{2}\mathbb{E}[\|\partial_1 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t) - \partial_1 f(\mathbf{w}_t, \mathbf{u}_t)\|_2^2].
\end{aligned}
$$

Hence,

$$
\begin{aligned}
&\mathbb{E}\left[\sum_{t=1}^{T}(\partial_1 f(\mathbf{w}_t,\mathbf{u}_t;\zeta_t)-\partial_1 f(\mathbf{w}_t,\mathbf{u}_t))^\top \mathbf{w}\right] \\
&+\mathbb{E}\left[\sum_{t=1}^{T}(\partial_2 f(\mathbf{w}_t,\mathbf{u}_t)-\partial_2 f(\mathbf{w}_t,\mathbf{u}_t;\zeta_t))^\top \mathbf{u}\right] \\
&\leq \frac{1}{2\eta_1}\|\hat{\mathbf{w}}_1-\mathbf{w}\|_2^2+\frac{1}{2\eta_2}\|\hat{\mathbf{u}}_t-\mathbf{u}\|_2^2+\frac{4\eta_1 G_1^2 T}{2}+\frac{4\eta_2 G_2^2 T}{2}.
\end{aligned} \tag{3.49}
$$

Combining (3.48) and (3.49), we have

$$
\mathbb{E}[\Delta(\bar{\mathbf{w}}_T,\bar{\mathbf{u}}_T)] \leq \frac{1}{\eta_1 T}\mathbb{E}[\|\mathbf{w}_1-\mathbf{w}\|_2^2]+\frac{1}{\eta_2 T}\mathbb{E}[\|\mathbf{u}_1-\mathbf{u}\|_2^2]+\frac{5\eta_1 G_1^2}{2}+\frac{5\eta_2 G_2^2}{2}.
$$

Hence, we conclude the proof. □

3.7 Stochastic Optimistic Mirror Prox

While simple in design, SGDA cannot enjoy a faster convergence when the function is smooth and the stochastic gradients have zero variance. A classical method to address this limitation is to use an extra-gradient. Let

$$
\mathbf{v}=\begin{bmatrix}\mathbf{w}\\ \mathbf{u}\end{bmatrix}, \quad \mathcal{M}(\mathbf{v})=\begin{bmatrix}\nabla_1 f(\mathbf{w},\mathbf{u})\\ -\nabla_2 f(\mathbf{w},\mathbf{u})\end{bmatrix}, \quad \mathcal{V}=\mathcal{W}\times\mathcal{U}.
$$

The extra-gradient method takes the following update with an initialization of $\mathbf{x}_1 \in \mathcal{V}$:

$$
\begin{aligned}
\mathbf{y}_t &= \arg\min_{\mathbf{v}\in\mathcal{V}} \mathcal{M}(\mathbf{x}_t)^\top \mathbf{v}+\frac{1}{2\eta}\|\mathbf{v}-\mathbf{x}_t\|_2^2 \\
\mathbf{x}_{t+1} &= \arg\min_{\mathbf{v}\in\mathcal{V}} \mathcal{M}(\mathbf{y}_t)^\top \mathbf{v}+\frac{1}{2\eta}\|\mathbf{v}-\mathbf{x}_t\|_2^2.
\end{aligned} \tag{3.50}
$$

The name "extragradient" comes from the fact that it uses two gradients $\mathcal{M}(\mathbf{x}_t)$ and $\mathcal{M}(\mathbf{y}_t)$ at each iteration.

The extragradient method can be generalized using the mirror descent steps with a Bregmand divergence $D_\varphi(\cdot,\cdot)$ defined by a strongly-convex function $\varphi : \mathcal{V} \to \mathbb{R}$:

$$
\begin{aligned}
\mathbf{y}_t &= \arg\min_{\mathbf{v}\in\mathcal{V}} \mathcal{M}(\mathbf{x}_t)^\top \mathbf{v}+\frac{1}{\eta}D_\varphi(\mathbf{v},\mathbf{x}_t) \\
\mathbf{x}_{t+1} &= \arg\min_{\mathbf{v}\in\mathcal{V}} \mathcal{M}(\mathbf{y}_t)^\top \mathbf{v}+\frac{1}{\eta}D_\varphi(\mathbf{v},\mathbf{x}_t).
\end{aligned} \tag{3.51}
$$

This method is called mirror prox.

Both methods can be extended to their stochastic versions. For example, the stochastic mirror prox method (SMP) uses the following update:

Algorithm 8 Stochastic Optimistic Mirror Prox (SOMP)

Input: learning rates η, starting points $\mathbf{x}_1 = \mathbf{y}_0 = (\mathbf{w}_1, \mathbf{u}_1)$
Compute $\mathbf{y}_1 = \arg\min_{\mathbf{v}\in\mathcal{V}} \mathcal{M}(\mathbf{y}_0;\zeta_0)^\top \mathbf{v} + \frac{1}{\eta} D_\varphi(\mathbf{v}, \mathbf{x}_1)$.
for $t = 1, \ldots, T$ **do**
    Compute unbiased gradient mapping $\mathcal{M}(\mathbf{y}_t;\zeta_t)$
    Update $\mathbf{x}_{t+1} = \arg\min_{\mathbf{v}\in\mathcal{V}} \mathcal{M}(\mathbf{y}_t;\zeta_t)^\top \mathbf{v} + \frac{1}{\eta} D_\varphi(\mathbf{v}, \mathbf{x}_t)$.
    Update $\mathbf{y}_{t+1} = \arg\min_{\mathbf{v}\in\mathcal{V}} \mathcal{M}(\mathbf{y}_t;\zeta_t)^\top \mathbf{v} + \frac{1}{\eta} D_\varphi(\mathbf{v}, \mathbf{x}_{t+1})$.
end for

$$\begin{aligned} \mathbf{y}_t &= \arg\min_{\mathbf{v}\in\mathcal{V}} \mathcal{M}(\mathbf{x}_t;\zeta_t')^\top \mathbf{v} + \frac{1}{\eta} D_\varphi(\mathbf{v}, \mathbf{x}_t) \\ \mathbf{x}_{t+1} &= \arg\min_{\mathbf{v}\in\mathcal{V}} \mathcal{M}(\mathbf{y}_t;\zeta_t)^\top \mathbf{v} + \frac{1}{\eta} D_\varphi(\mathbf{v}, \mathbf{x}_t), \end{aligned} \tag{3.52}$$

where $\mathbb{E}_\zeta[\mathcal{M}(\mathbf{x};\zeta)] = \mathcal{M}(\mathbf{x})$.

Stochastic Optimistic Mirror Prox: a variant with a Single Gradient Sequence

The updates of SMP (3.52) need to compute two stochastic gradient sequences $\{\mathcal{M}(\mathbf{x}_t, \zeta_t')\}$ and $\{\mathcal{M}(\mathbf{y}_t;\zeta_t)\}$, which double the costs of SGDA. A simple remedy is to use $\mathcal{M}(\mathbf{y}_{t-1};\zeta_{t-1})$ in the first update of $\mathbf{y}_t$, yielding

$$\begin{aligned} \mathbf{y}_t &= \arg\min_{\mathbf{v}\in\mathcal{V}} \mathcal{M}(\mathbf{y}_{t-1};\zeta_{t-1})^\top \mathbf{v} + \frac{1}{\eta} D_\varphi(\mathbf{v}, \mathbf{x}_t) \\ \mathbf{x}_{t+1} &= \arg\min_{\mathbf{v}\in\mathcal{V}} \mathcal{M}(\mathbf{y}_t;\zeta_t)^\top \mathbf{v} + \frac{1}{2\eta} D_\varphi(\mathbf{v}, \mathbf{x}_t). \end{aligned} \tag{3.53}$$

As a result, we only need to compute one sequence of stochastic gradients $\{\mathcal{M}(\mathbf{y}_t;\zeta_t)\}$. This method is known as stochastic optimistic mirror prox (SOMP).

Let us consider a special case when $\mathcal{V} = \mathbb{R}^d \times \mathbb{R}^{d'}$ and $D_\varphi(\mathbf{x}, \mathbf{y}) = \frac{1}{2}\|\mathbf{x} - \mathbf{y}\|_2^2$. The above update reduces to

$$\begin{aligned} \mathbf{y}_t &= \mathbf{x}_t - \eta \mathcal{M}(\mathbf{y}_{t-1};\zeta_{t-1}) \\ \mathbf{x}_{t+1} &= \mathbf{x}_t - \eta \mathcal{M}(\mathbf{y}_t;\zeta_t). \end{aligned} \tag{3.54}$$

This update can be re-written using one sequence of $\{\mathbf{y}_t\}$. By subtracting the second equation from the first one, we have

$$\mathbf{y}_t - \mathbf{x}_{t+1} = \eta \mathcal{M}(\mathbf{y}_t;\zeta_t) - \eta \mathcal{M}(\mathbf{y}_{t-1};\zeta_{t-1}). \tag{3.55}$$

As a result,

$$\begin{aligned} \mathbf{y}_t &= \mathbf{x}_{t+1} + \eta \mathcal{M}(\mathbf{y}_t;\zeta_t) - \eta \mathcal{M}(\mathbf{y}_{t-1};\zeta_{t-1}) \\ &= \mathbf{y}_{t+1} + \eta \mathcal{M}(\mathbf{y}_t;\zeta_t) + \eta \mathcal{M}(\mathbf{y}_t;\zeta_t) - \eta \mathcal{M}(\mathbf{y}_{t-1};\zeta_{t-1}). \end{aligned}$$

From this, we derive that

$$\mathbf{y}_{t+1} = \mathbf{y}_t - \eta(\mathcal{M}(\mathbf{y}_t;\zeta_t) + \mathcal{M}(\mathbf{y}_t;\zeta_t) - \mathcal{M}(\mathbf{y}_{t-1};\zeta_{t-1})). \tag{3.56}$$

This method applied to the min-max problem is known as stochastic optimistic gradient descent ascent (SOGDA), yielding the following primal and dual updates:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta(2\nabla_1 f(\mathbf{w}_t,\mathbf{u}_t;\zeta_t) - \nabla_1 f(\mathbf{w}_{t-1},\mathbf{u}_{t-1};\zeta_{t-1})) \tag{3.57}$$

$$\mathbf{u}_{t+1} = \mathbf{u}_t + \eta(2\nabla_2 f(\mathbf{w}_t,\mathbf{u}_t;\zeta_t) - \eta\nabla_2 f(\mathbf{w}_{t-1},\mathbf{u}_{t-1};\zeta_{t-1}))). \tag{3.58}$$

Convergence Analysis

We analyze the stochastic optimistic mirror prox method in Algorithm 8. We make the following assumption.

Assumption 3.11. *Suppose the following conditions hold:*

(i) $f(\mathbf{w},\mathbf{u})$ *is convex w.r.t* $\mathbf{w}$ *and concave w.r.t* $\mathbf{u}$.
(ii) Let $\varphi(\mathbf{z})$ *be a* α*-strongly convex function with respect to the norm* $\|\cdot\|$*, whose dual norm is denoted by* $\|\cdot\|_*$*,*
(ii) $\mathcal{M}(\mathbf{v})$ *is L-Lipschitz continuous such that*

$$\|\mathcal{M}(\mathbf{v}) - \mathcal{M}(\mathbf{v}')\|_*^2 \leq L^2\|\mathbf{v}-\mathbf{v}'\|^2.$$

(ii) There exist $\sigma_1,\sigma_2 > 0$ *such that*

$$\mathbb{E}_\zeta[\|\mathcal{M}(\mathbf{x};\zeta) - \mathcal{M}(\mathbf{x})\|_*^2] \leq \sigma^2, \forall \mathbf{x}\in\mathcal{V}.$$

(iii) $\max_{\mathbf{x}\in\mathcal{V},\mathbf{x}'\in\mathcal{V}} D_\varphi(\mathbf{x},\mathbf{x}') \leq D^2$.

Lemma 3.14 *Given* $\mathbf{x}$*, consider the updates:*

$$\begin{aligned}\mathbf{y} &= \arg\min_{\mathbf{v}\in\mathcal{V}} \gamma\mathcal{M}(\xi)^\top\mathbf{v} + D_\varphi(\mathbf{v},\mathbf{x}),\\ \mathbf{x}_+ &= \arg\min_{\mathbf{v}\in\mathcal{V}} \gamma\mathcal{M}(\zeta)^\top\mathbf{v} + D_\varphi(\mathbf{v},\mathbf{x}).\end{aligned} \tag{3.59}$$

For any $\mathbf{v}\in\mathcal{V}$*, we have*

$$\begin{aligned}\gamma\mathcal{M}(\zeta)^\top(\mathbf{y}-\mathbf{v}) \leq& D_\varphi(\mathbf{v},\mathbf{x}) - D_\varphi(\mathbf{v},\mathbf{x}_+) + \frac{\gamma^2}{\alpha}\|\mathcal{M}(\xi)-\mathcal{M}(\zeta)\|_*^2\\ &- \frac{\alpha}{2}[\|\mathbf{y}-\mathbf{x}\|^2 + \|\mathbf{y}-\mathbf{x}_+\|^2].\end{aligned} \tag{3.60}$$

Proof. First, by Lemma 3.8, we have

$$\|\mathbf{y}-\mathbf{x}_+\| \leq \frac{\gamma}{\alpha}\|\mathcal{M}(\zeta)-\mathcal{M}(\xi)\|_*. \tag{3.61}$$

Let $\phi(\mathbf{v}) = \gamma\mathcal{M}(\zeta)^\top(\mathbf{y}-\mathbf{v}) - D_\varphi(\mathbf{v},\mathbf{x}) + D_\varphi(\mathbf{v},\mathbf{x}_+)$. Given the optimality condition of $\mathbf{x}_+$, it is easy to verify that it also satisfies the optimality condition of $\max_{\mathbf{v}\in\mathcal{V}} \phi(\mathbf{v})$. As a result, $\phi(\mathbf{v}) \le \phi(\mathbf{x}_+), \forall \mathbf{v} \in \mathcal{V}$, i.e.,

$$\begin{aligned}
&\gamma\mathcal{M}(\zeta)^\top(\mathbf{y}-\mathbf{v}) - D_\varphi(\mathbf{v},\mathbf{x}) + D_\varphi(\mathbf{v},\mathbf{x}_+) \\
&\le \gamma\mathcal{M}(\zeta)^\top(\mathbf{y}-\mathbf{x}_+) - D_\varphi(\mathbf{x}_+,\mathbf{x}) \\
&= \gamma\mathcal{M}(\zeta)^\top(\mathbf{y}-\mathbf{x}_+) + \varphi(\mathbf{x}) + \nabla\varphi(\mathbf{x})^\top(\mathbf{x}_+-\mathbf{x}) - \varphi(\mathbf{x}_+) \\
&= \gamma(\mathcal{M}(\zeta)-\mathcal{M}(\xi))^\top(\mathbf{y}-\mathbf{x}_+) + \gamma\mathcal{M}(\xi)^\top(\mathbf{y}-\mathbf{x}_+) \\
&+ \varphi(\mathbf{x}) + \nabla\varphi(\mathbf{x})^\top(\mathbf{x}_+-\mathbf{x}) - \varphi(\mathbf{x}_+).
\end{aligned} \tag{3.62}$$

By the optimality condition of $\mathbf{y}$, for any $\mathbf{v} \in \mathcal{V}$ we have

$$(\gamma\mathcal{M}(\xi) + \nabla\varphi(\mathbf{y}) - \nabla\varphi(\mathbf{x}))^\top(\mathbf{y}-\mathbf{v}) \le 0$$

Plugging $\mathbf{v} = \mathbf{x}_+$ into the above inequality, we have

$$(\gamma\mathcal{M}(\xi) + \nabla\varphi(\mathbf{y}) - \nabla\varphi(\mathbf{x}))^\top(\mathbf{y}-\mathbf{x}_+) \le 0,$$

which implies that

$$\gamma\mathcal{M}(\xi)^\top(\mathbf{y}-\mathbf{x}_+) \le (\nabla\varphi(\mathbf{y}) - \nabla\varphi(\mathbf{x}))^\top(\mathbf{x}_+-\mathbf{y}).$$

Combining this with (3.62), we have

$$\begin{aligned}
&\gamma\mathcal{M}(\zeta)^\top(\mathbf{y}-\mathbf{v}) - D_\varphi(\mathbf{v},\mathbf{x}) + D_\varphi(\mathbf{v},\mathbf{x}_+) \le \gamma(\mathcal{M}(\zeta)-\mathcal{M}(\xi))^\top(\mathbf{y}-\mathbf{x}_+) \\
&+ (\nabla\varphi(\mathbf{y}) - \nabla\varphi(\mathbf{x}))^\top(\mathbf{x}_+-\mathbf{y}) + \varphi(\mathbf{x}) + \nabla\varphi(\mathbf{x})^\top(\mathbf{x}_+-\mathbf{x}) - \varphi(\mathbf{x}_+) \\
&= \gamma(\mathcal{M}(\zeta)-\mathcal{M}(\xi))^\top(\mathbf{y}-\mathbf{x}_+) \\
&+ \varphi(\mathbf{x}) + \nabla\varphi(\mathbf{x})^\top(\mathbf{y}-\mathbf{x}) - \varphi(\mathbf{x}_+) + (\nabla\varphi(\mathbf{y}))^\top(\mathbf{x}_+-\mathbf{y}) \\
&= \gamma(\mathcal{M}(\zeta)-\mathcal{M}(\xi))^\top(\mathbf{y}-\mathbf{x}_+) \\
&+ \varphi(\mathbf{x}) + \nabla\varphi(\mathbf{x})^\top(\mathbf{y}-\mathbf{x}) - \varphi(\mathbf{y}) + \varphi(\mathbf{y}) + (\nabla\varphi(\mathbf{y}))^\top(\mathbf{x}_+-\mathbf{y}) - \varphi(\mathbf{x}_+) \\
&= \gamma(\mathcal{M}(\zeta)-\mathcal{M}(\xi))^\top(\mathbf{y}-\mathbf{x}_+) - D_\varphi(\mathbf{y},\mathbf{x}) - D_\varphi(\mathbf{x}_+,\mathbf{y}) \\
&\le \frac{\gamma^2}{\alpha}\|\mathcal{M}(\zeta)-\mathcal{M}(\xi)\|_*^2 - \frac{\alpha}{2}\|\mathbf{y}-\mathbf{x}\|^2 - \frac{\alpha}{2}\|\mathbf{x}_+-\mathbf{y}\|^2,
\end{aligned}$$

where the last inequality uses (3.61) and the α-strong convexity of φ.

□

Theorem 3.14 *Let* $\bar{\mathbf{w}}_T = \frac{1}{T}\sum_{t=1}^T \mathbf{w}_t, \bar{\mathbf{u}}_T = \frac{1}{T}\sum_{t=1}^T \mathbf{u}_t$. *After* T *iterations, SOMP guarantees that*

$$\mathbb{E}[\Delta(\bar{\mathbf{w}}_T, \bar{\mathbf{u}}_T)] \le \frac{2D^2}{T\eta} + \frac{8\sigma^2\eta}{\alpha}.$$

If we set $\eta = \min(\frac{D}{2\sqrt{T}\sigma}, \frac{\alpha}{\sqrt{12}L})$, *we have*

$$\mathbb{E}[\Delta(\bar{\mathbf{w}}_T, \bar{\mathbf{u}}_T)] \leq O\left(\frac{LD^2}{T\alpha} + \frac{\sigma D}{\sqrt{T\alpha}}\right).$$

Why it matters

This result is consistent with the convergence of SGD for smooth convex minimization in Theorem 3.1. In particular, when $\sigma = 0$ (i.e., using the deterministic gradient), the convergence rate simplifies to $O(1/T)$.

Proof. Since the updates of $\mathbf{y}_t, \mathbf{x}_{t+1}$ follow that in (3.59), by applying Lemma 3.14, we have

$$\begin{aligned}
&\eta \mathcal{M}(\mathbf{y}_t, \zeta_t)^\top (\mathbf{y}_t - \mathbf{v}) \leq D_\varphi(\mathbf{v}, \mathbf{x}_t) - D_\varphi(\mathbf{v}, \mathbf{x}_{t+1}) \\
&+ \frac{\eta^2}{\alpha} \|\mathcal{M}(\mathbf{y}_t, \zeta_t) - \mathcal{M}(\mathbf{y}_{t-1}, \zeta_{t-1})\|_*^2 - \frac{\alpha}{2}[\|\mathbf{y}_t - \mathbf{x}_t\|^2 + \|\mathbf{y}_t - \mathbf{x}_{t+1}\|^2] \\
&\leq D_\varphi(\mathbf{v}, \mathbf{x}_t) - D_\varphi(\mathbf{v}, \mathbf{x}_{t+1}) \\
&+ \frac{\eta^2}{\alpha} \|\mathcal{M}(\mathbf{y}_t, \zeta_t) - \mathcal{M}(\mathbf{y}_{t-1}, \zeta_{t-1}) - \mathcal{M}(\mathbf{y}_t) + \mathcal{M}(\mathbf{y}_{t-1}) + (\mathcal{M}(\mathbf{y}_t) - \mathcal{M}(\mathbf{y}_{t-1}))\|_*^2 \\
&- \frac{\alpha}{2}[\|\mathbf{y}_t - \mathbf{x}_t\|^2 + \|\mathbf{y}_t - \mathbf{x}_{t+1}\|^2].
\end{aligned}$$

Let $\sigma_t^2 = \|\mathcal{M}(\mathbf{y}_t, \zeta_t) - \mathcal{M}(\mathbf{y}_t)\|_*^2$, then we have

$$\begin{aligned}
&\|\mathcal{M}(\mathbf{y}_t, \zeta_t) - \mathcal{M}(\mathbf{y}_{t-1}, \zeta_{t-1}) - \mathcal{M}(\mathbf{y}_t) + \mathcal{M}(\mathbf{y}_{t-1}) + (\mathcal{M}(\mathbf{y}_t) - \mathcal{M}(\mathbf{y}_{t-1})\|_*^2 \\
&\leq 3\|\mathcal{M}(\mathbf{y}_t, \zeta_t) - \mathcal{M}(\mathbf{y}_t)\|_*^2 + 3\|\mathcal{M}(\mathbf{y}_{t-1}, \zeta_{t-1}) - \mathcal{M}(\mathbf{y}_{t-1})\|_*^2 \\
&+ 3\|\mathcal{M}(\mathbf{y}_t) - \mathcal{M}(\mathbf{y}_{t-1})\|_*^2 \\
&\leq 3\sigma^2 + 3\sigma^2 + 3L^2\|\mathbf{y}_t - \mathbf{y}_{t-1}\|^2.
\end{aligned}$$

Combining the above two inequalities, we have

$$\begin{aligned}
&\eta \mathcal{M}(\mathbf{y}_t, \zeta_t)^\top (\mathbf{y}_t - \mathbf{v}) \leq D_\varphi(\mathbf{v}, \mathbf{x}_t) - D_\varphi(\mathbf{v}, \mathbf{x}_{t+1}) \\
&+ \frac{\eta^2}{\alpha}(6\sigma^2 + 3L^2\|\mathbf{y}_t - \mathbf{y}_{t-1}\|^2) - \frac{\alpha}{2}[\|\mathbf{y}_t - \mathbf{x}_t\|^2 + \|\mathbf{y}_t - \mathbf{x}_{t+1}\|^2].
\end{aligned}$$

Taking average over $t = 1, \ldots, T$, we have

$$\begin{aligned}
\frac{1}{T}\sum_{t=1}^T \mathcal{M}(\mathbf{y}_t)^\top (\mathbf{y}_t - \mathbf{v}) &\leq \frac{1}{T\eta} D_\varphi(\mathbf{v}, \mathbf{x}_1) \\
&+ \frac{\eta}{\alpha T}\sum_{t=1}^T (6\sigma^2 + 3L^2\|\mathbf{y}_t - \mathbf{y}_{t-1}\|^2) - \frac{\alpha}{2\eta T}\sum_{t=1}^T [\|\mathbf{y}_t - \mathbf{x}_t\|^2 + \|\mathbf{y}_t - \mathbf{x}_{t+1}\|^2] \\
&+ \frac{1}{T}\sum_{t=1}^T (\mathcal{M}(\mathbf{y}_t) - \mathcal{M}(\mathbf{y}_t, \zeta_t))^\top (\mathbf{y}_t - \mathbf{v}).
\end{aligned}$$

Let $\mathbf{y}_t = (\mathbf{w}_t, \mathbf{u}_t)$ and $\mathbf{v} = (\mathbf{w}, \mathbf{u}) = \arg\max_{\mathbf{w}\in\mathcal{W},\mathbf{u}\in\mathcal{U}} f(\bar{\mathbf{w}}_T, \mathbf{u}) - f(\mathbf{w}, \bar{\mathbf{u}}_T)$. We have

$$\begin{aligned}
&\frac{1}{T}\sum_{t=1}^{T}\mathcal{M}(\mathbf{y}_t)^\top(\mathbf{y}_t - \mathbf{v}) = \frac{1}{T}\sum_{t=1}^{T}(\nabla_1 f(\mathbf{w}_t, \mathbf{u}_t)^\top(\mathbf{w}_t - \mathbf{w}) - \nabla_2 f(\mathbf{w}_t, \mathbf{u}_t)^\top(\mathbf{u}_t - \mathbf{u}))\\
&\geq \frac{1}{T}\sum_{t=1}^{T}(f(\mathbf{w}_t, \mathbf{u}_t) - f(\mathbf{w}, \mathbf{u}_t) + f(\mathbf{w}_t, \mathbf{u}) - f(\mathbf{w}_t, \mathbf{u}_t))\\
&= \frac{1}{T}\sum_{t=1}^{T}(f(\mathbf{w}_t, \mathbf{u}) - f(\mathbf{w}, \mathbf{u}_t)) \geq f(\bar{\mathbf{w}}_T, \mathbf{u}) - f(\mathbf{w}, \bar{\mathbf{u}}_T).
\end{aligned}$$

As a result,

$$\begin{aligned}
&\Delta(\bar{\mathbf{w}}_T, \bar{\mathbf{u}}_T) \leq \frac{1}{T}\sum_{t=1}^{T}\mathcal{M}(\mathbf{y}_t)^\top(\mathbf{y}_t - \mathbf{v}) \leq \frac{1}{T\eta}D_\varphi(\mathbf{v}, \mathbf{x}_1)\\
&+ \frac{\eta}{\alpha T}\sum_{t=1}^{T}(6\sigma^2 + 3L^2\|\mathbf{y}_t - \mathbf{y}_{t-1}\|^2) - \frac{\alpha}{2\eta T}\sum_{t=1}^{T}[\|\mathbf{y}_t - \mathbf{x}_t\|^2 + \|\mathbf{y}_t - \mathbf{x}_{t+1}\|^2]\\
&+ \frac{1}{T}\sum_{t=1}^{T}(\mathcal{M}(\mathbf{y}_t) - \mathcal{M}(\mathbf{y}_t, \zeta_t))^\top(\mathbf{y}_t - \mathbf{v}).
\end{aligned}$$

The last term can be bounded by using Lemma 3.13. Define the virtual sequence with $\hat{\mathbf{y}}_1 = \mathbf{x}_1$:

$$\hat{\mathbf{y}}_{t+1} = \arg\min_{\mathbf{v}\in\mathcal{V}}(\mathcal{M}(\mathbf{y}_t) - \mathcal{M}(\mathbf{y}_t, \zeta_t))^\top\mathbf{v} + \frac{1}{\eta}D_\varphi(\mathbf{v}, \hat{\mathbf{y}}_t).$$

Then Lemma 3.13 implies that

$$\begin{aligned}
&\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}(\mathcal{M}(\mathbf{y}_t, \zeta_t) - \mathcal{M}(\mathbf{y}_t))^\top\mathbf{v}\right] \leq \mathbb{E}\left[\frac{1}{\eta T}D_\varphi(\mathbf{v}, \hat{\mathbf{y}}_1)\right]\\
&+ \mathbb{E}\left[\frac{\eta}{2\alpha T}\sum_{t=1}^{T}\|\mathcal{M}(\mathbf{y}_t) - \mathcal{M}(\mathbf{y}_t, \zeta_t)\|_*^2\right].
\end{aligned}$$

Combining the above results, we have

$$
\begin{aligned}
\mathbb{E}[\Delta(\bar{\mathbf{w}}_T, \bar{\mathbf{u}}_T)] &\leq \frac{2}{T\eta} D_\varphi(\mathbf{v}, \mathbf{x}_1) + \frac{8\sigma^2\eta}{\alpha} \\
&+ \mathbb{E}\left[\frac{3L^2\eta}{\alpha T}\sum_{t=1}^{T}\|\mathbf{y}_t - \mathbf{y}_{t-1}\|^2 - \frac{\alpha}{2\eta T}\sum_{t=1}^{T}[\|\mathbf{y}_t - \mathbf{x}_t\|^2 + \|\mathbf{y}_t - \mathbf{x}_{t+1}\|^2\right] \\
&\leq \frac{2}{T\eta} D_\varphi(\mathbf{v}, \mathbf{x}_1) + \frac{8\sigma^2\eta}{\alpha} + \mathbb{E}\left[\frac{3L^2\eta}{\alpha T}\sum_{t=1}^{T}[2\|\mathbf{y}_t - \mathbf{x}_t\|^2 + 2\|\mathbf{x}_t - \mathbf{y}_{t-1}\|^2]\right] \\
&- \mathbb{E}\left[\frac{\alpha}{2\eta T}\sum_{t=1}^{T}[\|\mathbf{y}_t - \mathbf{x}_t\|^2 + \|\mathbf{y}_t - \mathbf{x}_{t+1}\|^2\right].
\end{aligned}
$$

If $6L^2\frac{\eta}{\alpha} \leq \frac{\alpha}{2\eta}$, i.e., $\eta \leq \frac{\alpha}{\sqrt{12}L}$, the sum of the last two terms will be less than zero due to $\mathbf{x}_1 = \mathbf{y}_0$. Then, we have

$$
\mathbb{E}[\Delta(\bar{\mathbf{w}}_T, \bar{\mathbf{u}}_T)] \leq \frac{2}{T\eta} D_\varphi(\mathbf{v}, \mathbf{x}_1) + \frac{8\sigma^2\eta}{\alpha} \leq \frac{2D^2}{T\eta} + \frac{8\sigma^2\eta}{\alpha}.
$$

For the second part, optimizing the upper bound over η gives $\eta_* = \frac{D\sqrt{\alpha}}{2\sqrt{T}\sigma}$. If $\eta_* \leq \frac{\alpha}{\sqrt{12}L}$, i.e., $T \geq \frac{3D^2L^2}{\sigma^2\alpha}$, we set $\eta = \eta_*$, then

$$
\mathbb{E}\left[F(\bar{\mathbf{w}}_T) - F(\mathbf{w}_*)\right] \leq \frac{8\sigma D}{\sqrt{T\alpha}}.
$$

If $\eta_* > \frac{\alpha}{\sqrt{12}L}$, i.e., $\sigma^2 \leq \frac{3D^2L^2}{\alpha T}$, we set $\eta = \frac{\alpha}{\sqrt{12}L}$, then

$$
\mathbb{E}\left[\Delta(\bar{\mathbf{w}}_T, \bar{\mathbf{u}}_T)\right] \leq \frac{2\sqrt{12}LD^2}{T\alpha} + \frac{12LD^2}{\sqrt{3}T\alpha}.
$$

□

3.8 History and Notes

Stochastic Approximation and Mathematical Optimization

Stochastic approximation has a long history dating back to Robbins and Monro (1951) for solving a root finding problem $f(x) = \alpha$ using an iterative method $x_{t+1} = x_t - a_t(y_t - \alpha)$, where y_t is a stochastic variable such that $\mathbb{E}[y_t] = f(x_t)$. They studied the asymptotic convergence of $\lim_{t\to\infty}\mathbb{E}[(x_t - \theta)^2] = 0$ under some conditions, where θ is the solution to the root finding problem. It is notable that Herbert Robbins was regarded as one of the most influential mathematicians of the latter half of the 20th century, renowned for his seminal contributions to probability, algebra, and graph theory.

Inspired by Robbins and Monro (1951), Kiefer and Wolfowitz (1952) considered stochastic maximization of a regression function using a stochastic finite difference estimator of the gradient. Later, Chung (1954) established the convergence bound of Robbins-Monro's method under some conditions. Since then, the convergence of SGD has been extensively studied. Polyak and Juditsky (1992) analyzed the convergence of SGD with a simple averaging for stochastic optimization, which is sometimes referred to as Polyak-Juditsky averaging or Polyak averaging. This work assumes smoothness and strong convexity of the objective function and established a convergence rate of $O(1/T)$.

Nemirovski and Yudin (1978) is probably the first work that analyzes the non-asymptotic convergence of SGDA for general convex-concave min-max optimization without smoothness and strong convexity assumption. Their paper introduces the weighted averaging (weighted by the step size at each iteration) and establishes the convergence rate of $O(1/\sqrt{T})$. The optimal rate $O(1/T)$ for strongly-convex strongly-concave min-max problem was recently proved in Yan et al. (2020a).

The mirror descent method originates from Nemirovsky and Yudin (1983), which is also the work that is often cited for the lower bound of $O(1/\sqrt{T})$ for general convex problems. A more general form of SMD and its extension for convex-concave min-max problems using a Bregman divergence was later considered in (Nemirovski et al., 2009).

The non-asymptotic analysis of SGD for non-convex optimization was initiated by (Ghadimi and Lan, 2013). The non-asymptotic analysis of SGD for weakly convex optimization was developed by (Davis and Drusvyatskiy, 2019).

The proximal method dates back to the proximal point method proposed by Martinet (1972) and further developed in (Rockafellar, 1976). Lions and Mercier (1979) proposed a splitting method for finding a zero point of the sum of two maximal monotone operators. The forward backward splitting was first proposed by Pazy (1979) in the same context of finding a zero of sum of monotone operators. Its special instance for minimization problems known as projected gradient method was first studied by Goldstein (1964).

Coordinate descent has a long history in optimization, with its earliest roots traceable to the Gauss–Seidel iterations for solving linear systems in the 19th century. The method was later formalized and discussed in early optimization literature, including (Warga, 1963; Ortega and Rheinboldt, 1970; Luenberger, 1973). Rigorous analysis of convergence properties was developed in a sequence of influential works by Paul Tseng and others, including (Luo and Tseng, 1992; Tseng, 1990; Tseng and Bertsekas, 1987; Tseng, 2001). Recent developments of block coordinate descent including accelerated coordinate descent (Nesterov, 2012) and stochastic block coordinate descent (Dang and Lan, 2015).

The extragradient method was first proposed by Korpelevich (1976). The mirror prox method and its convergence rate $O(1/T)$ was proposed and established by Nemirovski (2004). The stochastic mirror prox method was analyzed in (Juditsky et al., 2011).

Optimization in machine learning

Frank Rosenblatt's pioneering work in the late 1950s introduced a learning rule for updating the Perceptron model (a single-layer neural network for binary classification) (Rosenblatt, 1962), a method that shares a conceptual foundation with modern stochastic gradient descent (SGD). The earliest instance of SGD for machine learning is perhaps the Widrow-Hoff algorithm (Widrow and Hoff, 1960) (also known as the least mean square' algorithm), which was used to train ADALINE - a single-layer neural network that produces a continuous output. Amari (1967) is the first work that applies SGD to optimize a neural network for binary and multi-class classification.

Starting in late 1980s, online prediction problem has attracted increasing attention in machine learning, whose developments have many parallels to stochastic optimization. Littlestone (1988) proposed the Winnow algorithm for learning Boolean functions. It was shown to be better than the earlier Perceptron learning algorithm in the sense that the number of mistakes grows only logarithmically with the number of irrelevant attributes in the examples. Later, it was generalized to weighted majority for learning with expert advice (Littlestone and Warmuth, 1994), and the exponentiated gradient method (Kivinen and Warmuth, 1997) for online learning with a decision variable from a simplex, which is a special case of SMD using the KL-divergence. It has impact on the development of Adaboost (Freund and Schapire, 1997).

During the first decade of the 21st century, online convex optimization emerged as a central topic in machine learning. A key focus was on regret bound analysis, which can be transferred into convergence guarantees for stochastic optimization via the online-to-batch conversion technique (Dekel and Singer, 2005). Regret bounds for online gradient descent were established for both convex loss functions (Zinkevich, 2003) and strongly convex loss functions (Hazan et al., 2007). The multi-epoch scheme for achieving an optimal rate of $O(1/T)$ for stochastic strongly convex optimization was established independently and concurrently in (Iouditski and Nesterov, 2010; Hazan and Kale, 2011; Ghadimi and Lan, 2012). Later, SGD has shown to be able to achieve the optimal rate for stochastic non-smooth strongly convex optimization using tail averaging (Rakhlin et al., 2012) or increased weighted averaging (Lacoste-Julien et al., 2012). The last iterate convergence of SGD for non-smooth convex optimization was established in (Shamir and Zhang, 2013).

The use of the ℓ_1 norm as a regularizer in the Lasso method was pioneered by Tibshirani (1996). The elastic net regularizer was later proposed in (Zou and Hastie, 2003), while the group lasso regularizer was introduced by (Yuan and Lin, 2006). More recently, the Piecewise Affine Regularizer (PAR) was proposed in (Jin et al., 2025). The nuclear norm minimization for promoting a low-rank matrix was first considered in (Fazel et al., 2001).

Pioneering works on the application of SGD to regularized empirical risk minimization in machine learning, including support vector machines, include (Zhang, 2004a; Shalev-Shwartz et al., 2007). The application of the proximal gradient method to ℓ_1 norm regularized problem was initiated by Daubechies et al. (2004), yielding an algorithm known as iterative thresholding. The application of SPGD to machine

learning with various regularization terms was studied in (Duchi and Singer, 2009). The application of SGD for optimizing truncated loss and its theory was studied in (Xu et al., 2019b).

The most famous application of coordinate descent methods in machine learning is the solver for support vector machine (Chang et al., 2008; Hsieh et al., 2008).

AdaGrad, proposed by Duchi et al. (2011), was a breakthrough in stochastic optimization for machine learning. It later inspired several popular stochastic algorithms for deep learning, including RMSprop (Hinton, 2018) and Adam (Kingma and Ba, 2015), which will be discussed in Chapter 6.

The first variant of stochastic optimistic mirror prox method appeared in the author's award-winning work (Chiang et al., 2012), inspired by Nemirovski's mirror prox method. It was introduced to address a long-standing challenge in online convex optimization for achieving variational regret bounds. This line of research later inspired the work of (Rakhlin and Sridharan, 2013), which formally coined the term optimistic mirror descent. More recently, stochastic optimistic mirror prox has been adopted for solving min–max problems in machine learning, including applications such as training generative adversarial networks (Daskalakis et al., 2018).

Discussion. The most important factor that affects the practical performance of SGD and other stochastic algorithms is the learning rate scheme η_t. In this chapter, we focus on a fixed learning rate $\eta_t = \eta$. However, it is usually not the best choice in practice. We can also develop theoretical analysis of these algorithms using decreasing learning rates, e.g., $\eta_t \propto 1/\sqrt{t}, 1/t$. However, these theoretical learning rate schemes are usually also not the best in practice. A practical approach is the step decay strategy as in Theorem 3.7, which gives a convergence that has only logarithmic dependence on the initial distance $\|\mathbf{w}_1 - \mathbf{w}_*\|_2$. This strategy also works for general stochastic convex optimization under generic error bound conditions in the form $\|\mathbf{w} - \mathbf{w}_*\|_2 \leq c(g(\mathbf{w}) - g(\mathbf{w}_*))^\theta$ with $\theta \in (0, 1]$ (Xu et al., 2017). Another issue of theoretical learning rates is that their best values that optimize the convergence bound may depend on some unknown parameters of the problem, e.g., $\mathbf{w}_*$, the smoothness constant, strong convexity modulus. This issue has triggered a line of research known as parameter-free algorithms (Orabona, 2019; Lan et al., 2023).

While this chapter focuses on classical stochastic methods that not only have important applications in machine learning but also significantly influence the approaches presented in later chapters, it does not cover several important algorithms, most notably accelerated gradient methods and their stochastic variants. These methods achieve optimal convergence rates for smooth convex objectives when the variance of stochastic gradients vanishes (Lan, 2012). For a comprehensive treatment of accelerated gradient methods, we refer to the textbook by Nesterov (2004), and for stochastic accelerated algorithms, we recommend Lan (2020). Variants of these methods will be introduced in Chapter 6.

Finally, I recommend the textbook (Recht and Wright, 2025), which provides a comprehensive treatment of convex optimization algorithms tailored for data analysis.

Chapter 4
Foundations: Stochastic Compositional Optimization

Abstract In this chapter, we introduce stochastic compositional optimization problems and their optimization algorithms, including stochastic compositional gradient descent and stochastic compositional momentum metholds. We also consider extensions of these techniques to structured optimization with compositional gradients including non-convex regularized problems, min-max optimization, min-min optimization and bilevel optimization. We focus on the complexity of these metholds for non-convex optimization.

Moving average is the core ingredient!

Contents

4.1 Stochastic Compositional Optimization

We have seen several advanced machine learning frameworks in the Chapter 2, including DRO, GDRO, EXM, and COCE. Unfortunately, existing stochastic gradient metholds such as SGD are not directly applicable to these new problems. The reason will become clear shortly. To address this challenge, we need new optimization tools.

In this chapter, we will consider a family of stochastic optimization problems called **stochastic compositional optimization (SCO)**, whose objective is given by

$$\min_{\mathbf{w}\in\mathbb{R}^d} F(\mathbf{w}) := \mathbb{E}_\xi f(\mathbb{E}_\zeta g(\mathbf{w};\zeta);\xi), \tag{4.1}$$

where ξ and ζ are random variables, $g(\cdot;\zeta) : \mathbb{R}^d \to \mathbb{R}^{d'}$ is the inner random function, and $f(\cdot;\xi) : \mathbb{R}^{d'} \to \mathbb{R}$ is the outer random function. Let $f(\cdot) = \mathbb{E}_\xi f(\cdot;\xi)$ and $g(\cdot) = \mathbb{E}_\zeta g(\cdot;\zeta)$. Then the objective function $F(\mathbf{w}) = f(g(\mathbf{w}))$ is a composition of two functions.

Examples

Example 4.1. *The KL-regularized DRO (2.14) is a special case of SCO by setting* $f(\cdot) = \lambda\log(\cdot)$ *and* $g(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^n \exp(\ell(\mathbf{w};\mathbf{x}_i,y_i)/\lambda)$.

Example 4.2. *The KL-constrained DRO (2.19) is a special case of SCO by setting* $\bar{g} = (g_1, g_2)$ $f(\bar{g}) = g_1\log(g_2) + g_1\rho$ *and* $g_1(\mathbf{w},\lambda) = \lambda, g_2(\mathbf{w},\lambda) = \frac{1}{n}\sum_{i=1}^n \exp(\ell(\mathbf{w};\mathbf{x}_i,y_i)/\lambda)$.

Example 4.3. *The compositional objective for AUC maximization (2.32) has a compositional term of* $f(g(\mathbf{w}))$*, where* $g(\mathbf{w})$ *is a stochastic function and* f *is a deterministic function.*

Optimization Challenge

The challenge of solving SCO lies in how to estimate the gradient $\nabla F(\mathbf{w}) = \nabla g(\mathbf{w})\nabla f(g(\mathbf{w}))$, where $\nabla g(\mathbf{w}) \in \mathbb{R}^{d\times d'}$ denotes the transpose of the Jacobian matrix of g at $\mathbf{w}$ and $\nabla f(g) \in \mathbb{R}^{d'}$ is a gradient of f at g.

A simple way of estimating the gradient is by using stochastic samples, i.e., $G(\mathbf{w};\xi,\zeta,\zeta') = \nabla g(\mathbf{w};\zeta)\nabla f(g(\mathbf{w};\zeta');\xi)$, where ξ,ζ,ζ' are random samples. One can also use mini-batch of random samples to compute the estimator. However, the problem is that $G(\mathbf{w};\xi,\zeta,\zeta')$ is a biased estimator when f is non-linear, i.e., $\mathbb{E}_{\xi,\zeta,\zeta'}G(\mathbf{w};\xi,\zeta,\zeta') \neq \nabla F(\mathbf{w})$. This will break all assumptions made in the convergence analysis in Chapter 3. Directly using this estimator in SGD could result in non-convergence or it requires a large batch size for estimating $g(\mathbf{w})$.

Algorithm 9 SCGD

Input: learning rate schedules $\{\eta_t\}_{t=1}^T$, $\{\gamma_t\}_{t=1}^T$; starting points $\mathbf{w}_1$, $\mathbf{u}_0$
for $t = 1, \ldots, T$ **do**
  Sample ζ_t, ζ_t' and ξ_t
  Compute the inner function value estimator $\mathbf{u}_t = (1-\gamma_t)\mathbf{u}_{t-1} + \gamma_t g(\mathbf{w}_t; \zeta_t)$
  Compute the vanilla gradient estimator $\mathbf{z}_t = \nabla g(\mathbf{w}_t; \zeta_t')\nabla f(\mathbf{u}_t; \xi_t)$
  Update the model $\mathbf{w}$ by $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{z}_t$
end for

4.2 Stochastic Compositional Gradient Descent

We assume both f and g are differentiable. Next, we introduce stochastic compositional gradient descent (SCGD) as a solution method for SCO. The key to the design is to track the sequence of $\{g(\mathbf{w}_t), t = 1, \ldots, T\}$ by a sequence of estimators $\{\mathbf{u}_t, t = 1, \ldots, T\}$. Let us consider the following problem:

$$\min_u \frac{1}{2}\|\mathbf{u} - g(\mathbf{w}_t)\|_2^2. \tag{4.2}$$

We compute $\mathbf{u}_t$ by using the SGD update:

$$\mathbf{u}_t = \mathbf{u}_{t-1} - \gamma_t(\mathbf{u}_{t-1} - g(\mathbf{w}_t; \zeta_t)) = (1-\gamma_t)\mathbf{u}_{t-1} + \gamma_t g(\mathbf{w}_t; \zeta_t), t \in [T], \tag{4.3}$$

where $g(\mathbf{w}; \zeta)$ is stochastic estimator of $g(\mathbf{w})$ such that $\mathbb{E}_\zeta[g(\mathbf{w}; \zeta)] = g(\mathbf{w})$. The update is also known as moving average sequence of $\{g(\mathbf{w}_t)\}$.

The intuition behind this is that when $\mathbf{w}_t$ converges (i.e., $\mathbf{w}_t - \mathbf{w}_{t-1} \to 0$), $\mathbf{u}_t$ is a better estimator of $g(\mathbf{w}_t)$ than $g(\mathbf{w}_t; \zeta_t)$. With $\mathbf{u}_t$, the gradient estimator can be computed by

$$\mathbf{z}_t = \nabla g(\mathbf{w}_t; \zeta_t')\nabla f(\mathbf{u}_t; \xi_t), \tag{4.4}$$

where ζ_t' is another independent random variable. Then, we can use it for updating $\mathbf{w}_t$:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{z}_t.$$

The detailed steps are presented in Algorithm 9.

Critical: Using ζ_t' instead of ζ_t in computing $\nabla g(\mathbf{w}_t; \zeta_t')$ is for simplicity of analysis, which decouple the dependence between $\mathbf{u}_t$ and ζ_t' as $\mathbf{u}_t$ depends on ζ_t. However, this will increase the number of random samples per-iteration. For practical implementation, one may just use $\zeta_t' = \zeta_t$.

4.2.1 Convergence Analysis

We make the following assumptions regarding the SCO problem (4.1).

Assumption 4.1. *There exist* $L_1, G_1 > 0$ *such that*

(i) f *is* L_1*-smooth, i.e.,* $\|\nabla f(g) - \nabla f(g')\|_2 \leq L_1 \|g - g'\|_2, \forall g, g'$*;*
(ii) $\mathbb{E}[\|\nabla f(g; \xi)\|_2^2] \leq G_1^2, \forall g$.

Assumption 4.2. *There exist* $G_2 > 0$ *such that* $\mathbb{E}[\|\nabla g(\mathbf{w}; \zeta)\|_2^2] \leq G_2^2, \forall \mathbf{w}$.

Due to Jensen's inequality, $\mathbb{E}[\|\nabla f(\cdot; \xi)\|_2^2] \leq G_1^2$, and $\mathbb{E}[\|\nabla g(\mathbf{w}; \zeta)\|_2^2] \leq G_2^2$ indicate the G_1-Lipschitz condition of f and G_2-Lipschitz condition of g, respectively.

Assumption 4.3. *There exist* $\sigma_0, \sigma_1, \sigma_2 > 0$ *such that*

(i) $\mathbb{E}[\|g(\mathbf{w}; \zeta) - g(\mathbf{w})\|_2^2] \leq \sigma_0^2, \forall \mathbf{w}$*;*
(ii) $\mathbb{E}[\|\nabla f(g; \xi) - \nabla f(g)\|_2^2] \leq \sigma_1^2, \quad \mathbb{E}[\|\nabla g(\mathbf{w}; \zeta) - \nabla g(\mathbf{w})\|_2^2] \leq \sigma_2^2, \forall \mathbf{w}, g$.
(iii) $F_* = \min_{\mathbf{w}} F(\mathbf{w}) > -\infty$.

Assumption 4.4. F *is* L_F*-smooth, i.e., there exist* $L_F > 0$ *such that* $\nabla F(\cdot)$ *is* L_F*-Lipschitz continuous.*

It is notable that the smoothness of F does not necessarily imply that g is smooth. One example is that if $g(\mathbf{w}) = \|\mathbf{w}\|_2$ and $f(g) = g^2$, the overall function $F(\mathbf{w}) = \|\mathbf{w}\|_2^2$ is smooth but the inner function g is non-smooth.

Lemma 4.1 *Under Assumptions 4.2 and 4.3(i), the* $\{\mathbf{u}_t\}_{t \geq 1}$ *sequence (4.3) satisfies that*

$$\mathbb{E}_{\zeta_t}\left[\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2\right] \leq (1 - \gamma_t)\|\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1})\|_2^2 + \gamma_t^2 \sigma_0^2 + \frac{G_2^2}{\gamma_t}\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2. \tag{4.5}$$

where $\mathbb{E}_{\zeta_t}$ *denotes the expectation over* ζ_t *given all previous randomness.*

Why it matters

The lemma admits an intuitive interpretation. The first term shows that $\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2$ is bounded by a contracting sequence. The second term is due to the noise in $g(\mathbf{w}_t; \zeta_t)$ and the third term is caused by the drifting from $\mathbf{w}_{t-1}$ to $\mathbf{w}_t$, both of which decay to zero under the conditions $\gamma_t^2 \to 0$ and $\frac{\mathbb{E}[\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2]}{\gamma_t} = O\left(\frac{\eta_{t-1}^2}{\gamma_t}\right) \to 0$, respectively.

Proof. In the following proof, we abuse the notation $\mathbb{E}_t$ to denote $\mathbb{E}_{\zeta_t}$. According to the update formula $\mathbf{u}_t = (1 - \gamma_t)\mathbf{u}_{t-1} + \gamma_t g(\mathbf{w}_t; \zeta_t)$ we have

$$\begin{aligned}\mathbb{E}_t\left[\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2\right] &= \mathbb{E}_t\left[\|(1-\gamma_t)\mathbf{u}_{t-1} + \gamma_t g(\mathbf{w}_t;\zeta_t) - g(\mathbf{w}_t)\|_2^2\right] \\ &= \mathbb{E}_t\left[\|(1-\gamma_t)(\mathbf{u}_{t-1} - g(\mathbf{w}_t)) + \gamma_t(g(\mathbf{w}_t;\zeta_t) - g(\mathbf{w}_t))\|_2^2\right].\end{aligned}$$

Note that $\mathbb{E}_t\left[(\mathbf{u}_{t-1} - g(\mathbf{w}_t))^\top(g(\mathbf{w}_t;\zeta_t) - g(\mathbf{w}_t))\right] = 0$. Thus,

$$\mathbb{E}_t\left[\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2\right] \le (1-\gamma_t)^2\|\mathbf{u}_{t-1} - g(\mathbf{w}_t)\|_2^2 + \gamma_t^2\sigma_0^2. \tag{4.6}$$

This inequality is same as Lemma 3.7 when we consider $\mathbf{u}_t$ as the SGD update for (4.2).

Due to the Young's inequality of inner product, we have $\|\mathbf{u}_{t-1} - g(\mathbf{w}_t)\|_2^2 \le (1+\alpha)\|\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1})\|_2^2 + (1+1/\alpha)\|g(\mathbf{w}_t) - g(\mathbf{w}_{t-1})\|_2^2$ for any $\alpha > 0$. Whence,

$$\begin{aligned}\mathbb{E}_t\left[\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2\right] \le& (1-\gamma_t)^2(1+\gamma_t)\|\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1})\|_2^2 \\ &+ (1-\gamma_t)^2(1+1/\gamma_t)G_2^2\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2 + \gamma_t^2\sigma_0^2.\end{aligned}$$

The proof is completed by noticing $(1-\gamma_t)^2(1+\gamma_t) \le 1-\gamma_t$ and $(1-\gamma_t)^2(1+1/\gamma_t) \le \frac{1}{\gamma_t}$. □

Lemma 4.2 *Under Assumptions 4.1, 4.2, 4.3 and 4.4 ,* `SCGD` *satisfies*

$$\begin{aligned}\mathbb{E}_{\zeta_t,\xi_t,\zeta_t'}\left[F(\mathbf{w}_{t+1})\right] \le& F(\mathbf{w}_t) - \frac{\eta_t}{2}\|\nabla F(\mathbf{w}_t)\|_2^2 + \frac{\eta_t G_2^2 L_1^2}{2}\mathbb{E}_{\zeta_t}\left[\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2\right] \\ &+ \frac{\eta_t^2 L_F G_1^2 G_2^2}{2}.\end{aligned} \tag{4.7}$$

Proof. In the following proof, we abuse the notation $\mathbb{E}_t$ to denote $\mathbb{E}_{\zeta_t,\xi_t,\zeta_t'}$. According to L_F-smoothness of F, we have

$$\begin{aligned}&F(\mathbf{w}_{t+1}) \le F(\mathbf{w}_t) + \nabla F(\mathbf{w}_t)^\top(\mathbf{w}_{t+1} - \mathbf{w}_t) + \frac{L_F}{2}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 \\ &= F(\mathbf{w}_t) - \eta_t\nabla F(\mathbf{w}_t)^\top\nabla g(\mathbf{w}_t;\zeta_t')\nabla f(\mathbf{u}_t;\xi_t) + \frac{\eta_t^2 L_F}{2}\left\|\nabla g(\mathbf{w}_t;\zeta_t')\nabla f(\mathbf{u}_t;\xi_t)\right\|_2^2.\end{aligned}$$

Then, we have

$$\begin{aligned}\mathbb{E}_t\left[F(\mathbf{w}_{t+1})\right] \le& F(\mathbf{w}_t) - \eta_t\|\nabla F(\mathbf{w}_t)\|_2^2 \\ &+ \eta_t\boxed{\mathbb{E}_t\left[\nabla F(\mathbf{w}_t)^\top(\nabla g(\mathbf{w}_t;\zeta_t')\nabla f(g(\mathbf{w}_t)) - \nabla g(\mathbf{w}_t;\zeta_t')\nabla f(\mathbf{u}_t))\right]} \\ &+ \frac{\eta_t^2 L_F}{2}\mathbb{E}_t\left[\left\|\nabla g(\mathbf{w}_t;\zeta_t')\nabla f(\mathbf{u}_t;\xi_t)\right\|_2^2\right],\end{aligned} \tag{4.8}$$

where we use the fact

$$\begin{aligned}&\mathbb{E}_{\zeta_t'}\left[\nabla g(\mathbf{w}_t;\zeta_t')\nabla f(g(\mathbf{w}_t))\right] = \nabla F(\mathbf{w}_t) \\ &\mathbb{E}_{\zeta_t,\zeta_t',\xi_t}\left[\nabla g(\mathbf{w}_t;\zeta_t')\nabla f(\mathbf{u}_t;\xi_t))\right] = \mathbb{E}_{\zeta_t,\zeta_t'}\left[\nabla g(\mathbf{w}_t;\zeta_t')\nabla f(\mathbf{u}_t))\right].\end{aligned}$$

Due to the Cauchy-Schwarz inequality and the Young's inequality of inner product, we have

$$\begin{aligned}
&\mathbb{E}_t\left[\nabla F(\mathbf{w}_t)^\top(\nabla g(\mathbf{w}_t;\zeta_t')\nabla f(g(\mathbf{w}_t)) - \nabla g(\mathbf{w}_t;\zeta_t')\nabla f(\mathbf{u}_t))\right] \\
&\leq \mathbb{E}_t\left[\frac{\|\nabla F(\mathbf{w}_t)\|_2^2\left\|\nabla g(\mathbf{w}_t;\zeta_t')\right\|_2^2}{2G_2^2}\right] + \mathbb{E}_{\zeta_t}\left[\frac{G_2^2}{2}\left\|\nabla f(g(\mathbf{w}_t)) - \nabla f(\mathbf{u}_t)\right\|_2^2\right] \\
&\leq \frac{\|\nabla F(\mathbf{w}_t)\|_2^2}{2} + \frac{G_2^2L_1^2}{2}\mathbb{E}_{\zeta_t}\left\|\mathbf{u}_t - g(\mathbf{w}_t)\right\|_2^2. \qquad (4.9)
\end{aligned}$$

For bounding the last term in (4.8), we proceed as follows:

$$\begin{aligned}
\boxed{\mathbb{E}_t\left[\left\|\nabla g(\mathbf{w}_t,\zeta_t')\nabla f(\mathbf{u}_t,\xi_t)\right\|_2^2\right]} &\leq \mathbb{E}_{\zeta_t,\zeta_t'}\left[\left\|\nabla g(\mathbf{w}_t;\zeta_t')\right\|_2^2\mathbb{E}_{\xi_t|\zeta_t,\zeta_t'}\left\|\nabla f(\mathbf{u}_t;\xi_t)\right\|_2^2\right] \\
&\leq G_1^2G_2^2. \qquad (4.10)
\end{aligned}$$

We finish the proof by plugging the last two inequalities into (4.8). □

Critical: We comment on the modifications required in the analysis when the same sample ζ_t is used to compute $\nabla g(\mathbf{w}_t;\zeta_t)$. In the original proof, there are two places highlighted in boxes, where we explicitly rely on the independence between $\mathbf{u}_t$ and ζ_t'. If instead we use the coupled estimator $\nabla g(\mathbf{w}_t;\zeta_t)\nabla f(\mathbf{u}_t;\xi_t)$, then the first term must be modified and bounded as follows:

$$\begin{aligned}
&\mathbb{E}_t\left[\nabla F(\mathbf{w}_t)^\top\left(\nabla g(\mathbf{w}_t;\zeta_t)\nabla f(g(\mathbf{w}_t);\xi_t) - \nabla g(\mathbf{w}_t;\zeta_t)\nabla f(\mathbf{u}_t;\xi_t)\right)\right] \\
&\leq \mathbb{E}_t\left[\frac{\|\nabla F(\mathbf{w}_t)\|^2\|\nabla g(\mathbf{w}_t;\zeta_t)\|^2}{2G_2^2}\right] + \mathbb{E}_t\left[\frac{G_2^2}{2}\|\nabla f(g(\mathbf{w}_t);\xi_t) - \nabla f(\mathbf{u}_t;\xi_t)\|^2\right].
\end{aligned}$$

To recover the same bound as in (4.9), we must impose a stronger regularity condition on f, namely,

$$\mathbb{E}_\xi\left[\|\nabla f(g;\xi) - \nabla f(g';\xi)\|^2\right] \leq L_1\|g - g'\|_2^2.$$

For the second boxed term, the corresponding expression becomes $\mathbb{E}_t\left[\|\nabla g(\mathbf{w}_t;\zeta_t)\nabla f(\mathbf{u}_t;\xi_t)\|^2\right]$, which in turn requires assuming that this quantity is uniformly bounded by a constant.

Combining Lemma 4.1 and Lemma 4.2, we can prove the following theorem of convergence for SCGD for a non-convex function.

Theorem 4.1 *Suppose Assumptions 4.1, 4.2, 4.3 and 4.4 hold. After T iterations of SCGD updates with parameters $\eta_t = \frac{\eta_1}{T^{3/5}}$, $\gamma_t = \frac{\gamma_1}{T^{2/5}}$, we have*

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\|\nabla F(\mathbf{w}_t)\|_2^2\right] \le \frac{2C_\Upsilon}{\eta_1 T^{2/5}} + \frac{L_1^2 G_1^2 G_2^6 \eta_1^2}{\gamma_1^2 T^{2/5}} + \frac{L_1^2 G_2^2 \sigma_0^2 \gamma_1}{T^{2/5}} + \frac{L_F G_1^2 G_2^2 \eta_1}{2T^{3/5}},$$

where $C_\Upsilon = F(\mathbf{w}_1) - F_* + \frac{L_1^2 C_2^2 \sigma_0^2}{2}\frac{\eta_1}{\gamma_1}$. *If* $\eta_t = \eta_1/t^{3/5}, \gamma_t = \gamma_1/t^{2/5}$, *then the convergence rate becomes* $O(\log T/T^{2/5})$.

Proof. Adding $\frac{L_1^2 G_2^2}{2}\frac{\eta_t}{\gamma_t}\mathbb{E}_t\left[\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2\right]$ on (4.7), we have

$$\begin{aligned}
&\mathbb{E}_t\left[F(\mathbf{w}_{t+1})\right] + \frac{L_1^2 G_2^2}{2}\frac{\eta_t}{\gamma_t}\mathbb{E}_t\left[\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2\right] \\
&\le F(\mathbf{w}_t) - \frac{\eta_t}{2}\|\nabla F(\mathbf{w}_t)\|_2^2 + (1+\gamma_t)\frac{\eta_t L_1^2 G_2^2}{2\gamma_t}\mathbb{E}_t\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2 + \frac{\eta_t^2 L_F G_1^2 G_2^2}{2}.
\end{aligned}$$

Applying Lemma 4.1 to bound the right hand side, we have

$$\begin{aligned}
&\mathbb{E}_t\left[F(\mathbf{w}_{t+1})\right] + \frac{L_1^2 G_2^2}{2}\frac{\eta_t}{\gamma_t}\mathbb{E}_t\left[\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2\right] \\
&\le F(\mathbf{w}_t) - \frac{\eta_t}{2}\|\nabla F(\mathbf{w}_t)\|_2^2 + (1-\gamma_t)(1+\gamma_t)\frac{L_1^2 G_2^2}{2}\frac{\eta_t}{\gamma_t}\|\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1})\|_2^2 \\
&+ \frac{(1+\gamma_t)L_1^2 G_2^2 G_2^2 \eta_t}{2\gamma_t^2}\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2 + \gamma_t\eta_t(1+\gamma_t)\frac{L_1^2 G_2^2 \sigma_0^2}{2} + \frac{\eta_t^2 L_F G_1^2 G_2^2}{2} \\
&\overset{\gamma_t \le 1}{\le} F(\mathbf{w}_t) + \frac{L_1^2 G_2^2}{2}\frac{\eta_t}{\gamma_t}\|\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1})\|_2^2 + \frac{\eta_t L_1^2 G_2^4}{\gamma_t^2}\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2 \\
&+ \gamma_t\eta_t L_1^2 G_2^2\sigma_0^2 + \frac{\eta_t^2 L_F G_1^2 G_2^2}{2} - \frac{\eta_t}{2}\|\nabla F(\mathbf{w}_t)\|_2^2.
\end{aligned}$$

We define the potential function $\Upsilon_t = F(\mathbf{w}_t) + \frac{L_1^2 G_2^2}{2}\frac{\eta_t}{\gamma_t}\|\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1})\|_2^2$. By the setting, we have $\frac{\eta_{t+1}}{\gamma_{t+1}} \le \frac{\eta_t}{\gamma_t}$, then

$$\Upsilon_{t+1} = F(\mathbf{w}_{t+1}) + \frac{L_1^2 G_2^2}{2}\frac{\eta_{t+1}}{\gamma_{t+1}}\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2 \le F(\mathbf{w}_{t+1}) + \frac{L_1^2 G_2^2}{2}\frac{\eta_t}{\gamma_t}\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2.$$

Then,

$$\begin{aligned}
\mathbb{E}_t\left[\Upsilon_{t+1}\right] \le \Upsilon_t &+ \frac{\eta_t L_1^2 G_2^4}{\gamma_t^2}\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2 + \gamma_t\eta_t L_1^2 G_2^2\sigma_0^2 + \frac{\eta_t^2 L_F G_1^2 G_2^2}{2} \\
&- \frac{\eta_t}{2}\|\nabla F(\mathbf{w}_t)\|_2^2.
\end{aligned}$$

Telescoping the above over $t = 1$ to T and use the tower property of conditional expectation.

$$\mathbb{E}\left[\sum_{t=1}^{T}\eta_t\left\|\nabla F(\mathbf{w}_t)\right\|_2^2\right] \leq 2\mathbb{E}\left[\Upsilon_1-\Upsilon_{T+1}\right]+2L_1^2G_2^4\sum_{t=1}^{T}\gamma_t^{-2}\eta_t\eta_{t-1}^2G_1^2G_2^2 + L_1^2G_2^2\sigma_0^2\sum_{t=1}^{T}\gamma_t\eta_t+\frac{L_FG_1^2G_2^2}{2}\sum_{t=1}^{T}\eta_t^2.$$

where we use the fact $\mathbb{E}[\|\mathbf{w}_t-\mathbf{w}_{t-1}\|_2^2] = \mathbb{E}[\eta_{t-1}^2\|\nabla g(\mathbf{w}_t;\zeta_t')\nabla f(\mathbf{u}_t;\xi_t)\|_2^2] \leq \eta_{t-1}^2G_1^2G_2^2$. Let $\mathbf{w}_0=\mathbf{w}_1$ and $\mathbf{u}_0=g(\mathbf{w}_0;\zeta_1)$. Then, we have

$$\begin{aligned}\mathbb{E}\left[\Upsilon_1-\Upsilon_{T+1}\right] &\leq \mathbb{E}\left[F(\mathbf{w}_1)+\frac{L_1^2C_2^2}{2}\frac{\eta_1}{\gamma_1}\left\|\mathbf{u}_0-g(\mathbf{w}_0)\right\|_2^2\right]-F_*\\ &\leq F(\mathbf{w}_1)-F_*+\frac{L_1^2G_2^2\sigma_0^2}{2}\frac{\eta_1}{\gamma_1}.\end{aligned}$$

We define $C_\Upsilon = F(\mathbf{w}_1)-F_*+\frac{L_1^2G_2^2\sigma_0^2}{2}\frac{\eta_1}{\gamma_1}$. Then we have

$$\mathbb{E}\left[\left\|\nabla F(\mathbf{w}_\tau)\right\|_2^2\right] \leq \frac{2C_\Upsilon}{\sum_{t=1}^{T}\eta_t}+L_1^2G_2^6G_1^2\frac{\sum_{t=1}^{T}\gamma_t^{-2}\eta_t\eta_{t-1}^2}{\sum_{t=1}^{T}\eta_t} + L_1^2G_2^2\sigma_0^2\frac{\sum_{t=1}^{T}\gamma_t\eta_t}{\sum_{t=1}^{T}\eta_t}+\frac{L_FG_1^2G_2^2}{2}\frac{\sum_{t=1}^{T}\eta_t^2}{\sum_{t=1}^{T}\eta_t}.$$

Plugging the constant values of $\eta_t=\frac{\eta_1}{T^{3/5}}$ and $\gamma_t=\frac{\gamma_1}{T^{2/5}}$, we have

$$\mathbb{E}\left[\left\|\nabla F(\mathbf{w}_\tau)\right\|_2^2\right] \leq \frac{2C_\Upsilon}{\eta_1T^{2/5}}+\frac{L_1^2G_1^2G_2^6\eta_1^2}{\gamma_1^2T^{2/5}}+\frac{L_1^2G_2^2\sigma_0^2\gamma_1}{T^{2/5}}+\frac{L_FG_1^2G_2^2\eta_1}{2T^{3/5}}.$$

If $\eta_t=O(1/t^{3/5})$, $\gamma_t=O(1/t^{2/5})$, $\frac{\eta_{t+1}}{\gamma_{t+1}}\leq\frac{\eta_t}{\gamma_t}$ is satisfied. Besides, we have $\sum_{t=1}^{T}\eta_t = O(T^{2/5})$, $\sum_{t=1}^{T}\eta_t^2=O(1)$, $\sum_{t=1}^{T}\gamma_t\eta_t=O(\log T)$, $\sum_{t=1}^{T}\gamma_t^{-2}\eta_t\eta_{t-1}^2=O(\log T)$. Then, we have $\mathbb{E}\left[\left\|\nabla F(\mathbf{w}_\tau)\right\|_2^2\right]\leq\tilde{O}(1/T^{2/5})$. □

4.2.2 An Improved Complexity with Smooth Inner Function

If we replace the smoothness assumption of F by the smoothness of g, we can establish a better complexity of SCGD.

Assumption 4.5. *g is L_2-smooth, i.e., there exist $L_2>0$ such that $\nabla g(\cdot)$ is L_2-Lipschitz continuous.*

Assumptions 4.1 and 4.5 ensures that F is smooth.

Lemma 4.3 *Under Assumptions 4.1 and 4.5, we have F is L_F-smooth, where $L_F = G_1L_2+G_2^2L_1$.*

Proof. Since $\nabla F(\mathbf{w}) = \nabla g(\mathbf{w})\nabla f(g(\mathbf{w}))$, we have

$$\begin{aligned}
&\|\nabla g(\mathbf{w}_1)\nabla f(g(\mathbf{w}_1)) - \nabla g(\mathbf{w}_2)\nabla f(g(\mathbf{w}_2))\|_2 \\
&= \|\nabla g(\mathbf{w}_1)\nabla f(g(\mathbf{w}_1)) - \nabla g(\mathbf{w}_1)\nabla f(g(\mathbf{w}_2)) \\
&\quad + \nabla g(\mathbf{w}_1)\nabla f(g(\mathbf{w}_2)) - \nabla g(\mathbf{w}_2)\nabla f(g(\mathbf{w}_2))\|_2 \\
&\leq G_2^2 L_1 \|\mathbf{w}_1 - \mathbf{w}_2\|_2 + G_1 L_2 \|\mathbf{w}_1 - \mathbf{w}_2\|_2.
\end{aligned}$$

□

Lemma 4.4 *Let* $\mathbf{z}_t = \nabla g(\mathbf{w}_t; \zeta_t')\nabla f(\mathbf{u}_t; \xi_t)$, $\mathcal{M}_t = \mathbb{E}_t[\mathbf{z}_t]$. *Then*

$$\begin{aligned}
&\mathbb{E}_t[\|\mathbf{z}_t - \mathcal{M}_t\|_2^2] \leq G_1^2\sigma_2^2 + G_2^2\sigma_1^2, \\
&\mathbb{E}_t[\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2] \leq \eta_t^2 G_1^2 G_2^2, \\
&\mathbb{E}_t[\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2] \leq \eta_t^2\|\mathcal{M}_t\|_2^2 + \eta_t^2(G_1^2\sigma_2^2 + G_2^2\sigma_1^2).
\end{aligned}$$

where $\mathbb{E}_t$ *denotes* $\mathbb{E}_{\zeta_t', \xi_t}$ *conditioned on* $\mathbf{w}_t, \mathbf{u}_t$.

Proof. First, we have

$$\begin{aligned}
&\mathbb{E}_t[\|\mathbf{z}_t - \mathcal{M}_t\|_2^2] = \mathbb{E}_t[\left\|\nabla g(\mathbf{w}_t; \zeta_t')\nabla f(\mathbf{u}_t; \xi_t) - \nabla g(\mathbf{w}_t)\nabla f(\mathbf{u}_t)\right\|_2^2] \\
&= \mathbb{E}_t[\|\nabla g(\mathbf{w}_t)\nabla f(\mathbf{u}_t) - \nabla g(\mathbf{w}_t)\nabla f(\mathbf{u}_t; \xi_t) \\
&+ \nabla g(\mathbf{w}_t)\nabla f(\mathbf{u}_t; \xi_t) - \nabla g(\mathbf{w}_t; \zeta_t')\nabla f(\mathbf{u}_t; \xi_t)\|_2^2] \\
&\leq G_2^2\sigma_1^2 + G_1^2\sigma_2^2.
\end{aligned}$$

Next, due to Assumption 4.1, 4.2 we have

$$\mathbb{E}_t[\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2] = \mathbb{E}_t[\eta_t^2\|\nabla g(\mathbf{w}_t; \zeta_t')\nabla f(\mathbf{u}_t; \xi_t)\|_2^2] \leq \eta_t^2 G_1^2 G_2^2.$$

Second, we have

$$\mathbb{E}_t[\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2] = \mathbb{E}_t[\eta_t^2\|\mathbf{z}_t - \mathcal{M}_t + \mathcal{M}_t\|_2^2] = \mathbb{E}_t[\eta_t^2\|\mathbf{z}_t - \mathcal{M}_t\|_2^2] + \eta_t^2\|\mathcal{M}_t\|_2^2.$$

Plugging the first result into the above, we finish the proof. □

Next, we develop two lemmas similar to Lemma 4.1 and Lemma 4.2.

Lemma 4.5 *Under Assumptions 4.2, 4.3 and 4.5, if* $\eta_{t-1}^2 \leq \frac{\gamma_t}{L_2^2 G_1^2}$ *then the* $\{\mathbf{u}_t\}_{t\geq 1}$ *sequence (4.3) satisfies that*

$$\begin{aligned}
\mathbb{E}\left[\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2\right] &\leq (1-\gamma_t)\mathbb{E}[\|\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1})\|_2^2] + \frac{4\eta_{t-1}^2 G_2^2}{\gamma_t}\mathbb{E}[\|\mathcal{M}_{t-1}\|_2^2] \\
&\quad + \gamma_t^2\sigma_0^2 + \frac{3\eta_{t-1}^2 G_2^2}{2}(G_1^2\sigma_2^2 + G_2^2\sigma_1^2). \qquad (4.11)
\end{aligned}$$

Proof. Similar to the proof of Lemma 4.1, we have

$$\mathbb{E}_t\left[\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2\right] \leq (1-\gamma_t)^2\|\mathbf{u}_{t-1} - g(\mathbf{w}_t)\|_2^2 + \gamma_t^2\sigma_0^2. \tag{4.12}$$

Next, we will handle $\|\mathbf{u}_{t-1} - g(\mathbf{w}_t)\|_2^2$ differently by using the smoothness of g.

$$\begin{aligned}
\|\mathbf{u}_{t-1} - g(\mathbf{w}_t)\|_2^2 &= \|\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1}) + g(\mathbf{w}_{t-1}) - g(\mathbf{w}_t)\|_2^2 \\
&= \|\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1})\|_2^2 + \|g(\mathbf{w}_{t-1}) - g(\mathbf{w}_t)\|_2^2 \\
&+ (\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1}))^\top(g(\mathbf{w}_{t-1}) - g(\mathbf{w}_t)) \\
&\leq \|\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1})\|_2^2 + G_2^2\|\mathbf{w}_{t-1} - \mathbf{w}_t\|_2^2 \\
&+ 2(\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1}))^\top(g(\mathbf{w}_{t-1}) - g(\mathbf{w}_t)).
\end{aligned}$$

Taking expectation on both sides and applying Lemma 4.4, we have

$$\begin{aligned}
&\mathbb{E}[\|\mathbf{u}_{t-1} - g(\mathbf{w}_t)\|_2^2] \leq \mathbb{E}[\|\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1})\|_2^2] + \eta_{t-1}^2 G_2^2\mathbb{E}[\|\mathcal{M}_{t-1}\|_2^2] \\
&+ \eta_{t-1}^2 G_2^2(G_2^2\sigma_1^2 + G_1\sigma_2^2) + \mathbb{E}[2(\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1}))^\top(g(\mathbf{w}_{t-1}) - g(\mathbf{w}_t))].
\end{aligned}$$

Instead of using the Young's inequality of inner product to bound the last term, we proceed as follows:

$$\begin{aligned}
&\mathbb{E}[(\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1}))^\top(g(\mathbf{w}_{t-1}) - g(\mathbf{w}_t))] \\
&= \underbrace{\mathbb{E}[(\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1}))^\top\nabla g(\mathbf{w}_{t-1})^\top(\mathbf{w}_{t-1} - \mathbf{w}_t)]}_{A} \\
&+ \underbrace{\mathbb{E}[(\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1}))^\top(g(\mathbf{w}_{t-1}) - g(\mathbf{w}_t) + \nabla g(\mathbf{w}_{t-1})^\top(\mathbf{w}_t - \mathbf{w}_{t-1}))]}_{B}.
\end{aligned}$$

To bound A, we have

$$\begin{aligned}
A &= \mathbb{E}[(\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1}))^\top\nabla g(\mathbf{w}_{t-1})^\top\eta_{t-1}\mathcal{M}_{t-1}] \\
&\leq \mathbb{E}[\alpha_t\|(\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1}))^\top\|^2 + \frac{\eta_{t-1}^2}{4\alpha_t}\|\nabla g(\mathbf{w}_{t-1})^\top\mathcal{M}_{t-1}\|_2^2] \\
&\leq \mathbb{E}[\alpha_t\|(\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1}))^\top\|^2 + \frac{\eta_{t-1}^2 G_2^2}{4\alpha_t}\|\mathcal{M}_{t-1}\|_2^2].
\end{aligned}$$

To bound B, we have

$$\begin{aligned}
B &\leq \mathbb{E}[\|\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1})\|_2\|g(\mathbf{w}_{t-1}) - g(\mathbf{w}_t) + \nabla g(\mathbf{w}_{t-1})^\top(\mathbf{w}_t - \mathbf{w}_{t-1})\|_2] \\
&\leq \mathbb{E}[\|\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1})\|_2\frac{L_2}{2}\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2] \\
&\leq \frac{L_2^2}{4G_2^2}\mathbb{E}[\|\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1})\|_2^2\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2] + \frac{G_2^2}{4}\mathbb{E}[\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2],
\end{aligned}$$

where the first inequality uses the smoothness of g and the last inequality uses the Young's inequality. To proceed, we utilize the first bound of $\mathbb{E}_{t-1}[\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2]$

in lemma 4.4 to bound the first term, and utilize its second bound in lemma 4.4 to bound the second $\mathbb{E}[\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2]$. Thus, we have

$$B \leq \frac{\eta_{t-1}^2 L_2^2 G_1^2}{4}\mathbb{E}[\|\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1})\|_2^2] + \frac{\eta_{t-1}^2 G_2^2}{4}\mathbb{E}[\|\mathcal{M}_{t-1}\|_2^2] + \frac{\eta_{t-1}^2 G_2^2}{4}(G_1^2\sigma_2^2 + G_2^2\sigma_1^2)].$$

Combing the bounds for A and B, we have

$$\begin{aligned}&\mathbb{E}[(\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1}))^\top (g(\mathbf{w}_{t-1}) - g(\mathbf{w}_t))] \\ &= \left(\alpha_t + \frac{\eta_{t-1}^2 L_2^2 G_1^2}{4}\right)\mathbb{E}[\|\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1})\|_2^2] + \left(\frac{\eta_{t-1}^2 G_2^2}{4\alpha_t} + \frac{\eta_{t-1}^2 G_2^2}{4}\right)\mathbb{E}[\|\mathcal{M}_{t-1}\|_2^2] \\ &+ \frac{\eta_{t-1}^2 G_2^2}{4}(G_1^2\sigma_2^2 + G_2^2\sigma_1^2).\end{aligned}$$

As a result,

$$\begin{aligned}\mathbb{E}[\|\mathbf{u}_{t-1} - g(\mathbf{w}_t)\|_2^2] &\leq \left(1 + 2\alpha_t + \frac{\eta_{t-1}^2 L_2^2 G_1^2}{2}\right)\|\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1})\|_2^2 \\ &+ \left(\eta_{t-1}^2 G_2^2 + \frac{\eta_{t-1}^2 G_2^2}{2\alpha_t} + \frac{\eta_{t-1}^2 G_2^2}{2}\right)\mathbb{E}[\|\mathcal{M}_{t-1}\|_2^2] \\ &+ \left(\eta_{t-1}^2 G_2^2 + \frac{\eta_{t-1}^2 G_2^2}{2}\right)(G_1^2\sigma_2^2 + G_2^2\sigma_1^2).\end{aligned}$$

We let $\alpha_t = \frac{\gamma_t}{4} < 1$, $\frac{\eta_{t-1}^2 L_2^2 G_1^2}{2} \leq \frac{\gamma_t}{2}$. Combining the above inequality with (4.12), we can finish the proof.

□

Lemma 4.6 *Under Assumptions 4.1, 4.2, 4.3 and 4.5, if $\eta_t L_F \leq 1/4$ then SCGD satisfies*

$$\begin{aligned}\mathbb{E}[F(\mathbf{w}_{t+1})] \leq &\mathbb{E}\left[F(\mathbf{w}_t) - \frac{\eta_t}{2}\|\nabla F(\mathbf{w}_t)\|_2^2 - \frac{\eta_t}{4}\|\mathcal{M}_t\|_2^2\right] \\ &+ \frac{\eta_t G_2^2 L_1^2}{2}\mathbb{E}[\|g(\mathbf{w}_t) - \mathbf{u}_t\|_2^2] + 2\eta_t^2 L_F(G_2^2\sigma_1^2 + G_1^2\sigma_2^2).\end{aligned} \tag{4.13}$$

Proof. According to Lemma 4.3 (L_F-smoothness of F), we have

$$\begin{aligned}&F(\mathbf{w}_{t+1}) \leq F(\mathbf{w}_t) + \nabla F(\mathbf{w}_t)^\top(\mathbf{w}_{t+1} - \mathbf{w}_t) + \frac{L_F}{2}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 \\ =&F(\mathbf{w}_t) - \eta_t \nabla F(\mathbf{w}_t)^\top \nabla g(\mathbf{w}_t; \zeta_t')\nabla f(\mathbf{u}_t; \xi_t) + \frac{\eta_t^2 L_F}{2}\left\|\nabla g(\mathbf{w}_t; \zeta_t')\nabla f(\mathbf{u}_t; \xi_t)\right\|_2^2.\end{aligned}$$

Taking expectation on both sides, we have

$$\mathbb{E}[F(\mathbf{w}_{t+1})] \leq \mathbb{E}[F(\mathbf{w}_t)] - \eta_t \mathbb{E}[\nabla F(\mathbf{w}_t)^\top \mathcal{M}_t] + \frac{\eta_t^2 L_F}{2} \mathbb{E}[\|\mathbf{z}_t - \mathcal{M}_t + \mathcal{M}_t\|_2^2]$$
$$= \mathbb{E}[F(\mathbf{w}_t)] - \eta_t \mathbb{E}[\nabla F(\mathbf{w}_t)^\top \mathcal{M}_t] + \eta_t^2 L_F \mathbb{E}[\|\mathbf{z}_t - \mathcal{M}_t\|_2^2] + \eta_t^2 L_F \mathbb{E}[\|\mathcal{M}_t\|_2^2]$$

Using $-2\mathbf{a}^\top \mathbf{b} = \|\mathbf{a} - \mathbf{b}\|_2^2 - \|\mathbf{a}\|_2^2 - \|\mathbf{b}\|_2^2$, we have

$$\mathbb{E}[F(\mathbf{w}_{t+1})] \leq \mathbb{E}[F(\mathbf{w}_t) - \frac{\eta_t}{2}\|\nabla F(\mathbf{w}_t)\|_2^2 - \frac{\eta_t}{2}\|\mathcal{M}_t\|_2^2]$$
$$+ \frac{\eta_t}{2}\mathbb{E}[\|\nabla F(\mathbf{w}_t) - \mathcal{M}_t\|_2^2] + \eta_t^2 L_F \mathbb{E}[\|\mathbf{z}_t - \mathcal{M}_t\|_2^2] + \eta_t^2 L_F \mathbb{E}[\|\mathcal{M}_t\|_2^2].$$

Next, we bound $\mathbb{E}[\|\nabla F(\mathbf{w}_t) - \mathcal{M}_t\|_2^2]$.

$$\mathbb{E}[\|\nabla F(\mathbf{w}_t) - \mathcal{M}_t\|_2^2] = \mathbb{E}[\|\nabla g(\mathbf{w}_t)\nabla f(g(\mathbf{w}_t)) - \nabla g(\mathbf{w}_t)\nabla f(\mathbf{u}_t)\|_2^2]$$
$$\leq G_2^2 L_1^2 \mathbb{E}[\|g(\mathbf{w}_t) - \mathbf{u}_t\|_2^2].$$

Combining the above inequalities, we have

$$\mathbb{E}[F(\mathbf{w}_{t+1})] \leq \mathbb{E}[F(\mathbf{w}_t) - \frac{\eta_t}{2}\|\nabla F(\mathbf{w}_t)\|_2^2 - \frac{\eta_t}{2}\|\mathcal{M}_t\|_2^2]$$
$$+ \frac{\eta_t G_2^2 L_1^2}{2}\mathbb{E}[\|g(\mathbf{w}_t) - \mathbf{u}_t\|_2^2] + \eta_t^2 L_F (G_2^2\sigma_1^2 + G_1^2\sigma_2^2) + \eta_t^2 L_F \mathbb{E}[\|\mathcal{M}_t\|_2^2].$$

If $\eta_t L_F \leq 1/4$, we have $-\frac{\eta_t}{2}\|\mathcal{M}_t\|_2^2 + \eta_t^2 L_F \|\mathcal{M}_t\|_2^2 \leq \frac{\eta_t}{4}\|\mathcal{M}_t\|_2^2$, which concludes the proof. □

Finally, we establish the following convergence of SCGD under the smoothness condition of g.

Theorem 4.2 *Suppose Assumptions 4.1, 4.5 and 4.3 hold. Run SCGD with T iterations with parameters $\eta_t = \frac{\eta_1}{\sqrt{T}}, \gamma_t = \frac{\gamma_1}{\sqrt{T}}$, where $\eta_1 \leq \min(\frac{\gamma_1}{\sqrt{8G_2^2 L_1}}, \frac{\sqrt{2\gamma_1}}{L_2 G_1}, \frac{1}{4L_F})$. Then we have*

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\|\nabla F(\mathbf{w}_t)\|_2^2\right] \leq O\left(\frac{C_\Upsilon}{\eta_1\sqrt{T}} + \frac{L_1\gamma_1^2\sigma_0^2}{\eta_1\sqrt{T}} + \frac{\eta_1(L_F + L_1G_2^2)(G_2^2\sigma_1^2 + G_1^2\sigma_2^2)}{\sqrt{T}}\right),$$

where $C_\Upsilon = F(\mathbf{w}_1) - F_* + \frac{L_1}{\sqrt{6}}\|\mathbf{u}_1 - g(\mathbf{w}_1)\|_2^2$.

Why it matters

From Theorem 4.2, we can derive that in order to find an ϵ-level stationary solution of a smooth non-convex compositional function (whose gradient norm is less than ϵ), SCGD needs a sample complexity of $O(\frac{L_1^4}{\epsilon^4})$. The order in terms of ϵ is the same order as that of SGD for solving non-convex ERM.

Proof. By Lemma 4.5, and Lemma 4.6, we have

$$\mathbb{E}[F(\mathbf{w}_{t+1})] \leq \mathbb{E}[F(\mathbf{w}_t) - \frac{\eta_t}{2}\|\nabla F(\mathbf{w}_t)\|_2^2 - \frac{\eta_t}{4}\|\mathcal{M}_t\|_2^2]$$
$$+ \frac{\eta_t G_2^2 L_1^2}{2}\mathbb{E}[\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2] + \eta_t^2 L_F(G_2^2\sigma_1^2 + G_1^2\sigma_2^2),$$
$$\mathbb{E}\left[\|\mathbf{u}_{t+1} - g(\mathbf{w}_{t+1})\|_2^2\right] \leq (1-\gamma_{t+1})\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2 + \frac{4\eta_t^2 G_2^2}{\gamma_{t+1}}\mathbb{E}[\|\mathcal{M}_t\|_2^2]$$
$$+ \gamma_{t+1}^2\sigma_0^2 + \frac{3\eta_t^2 G_2^2}{2}(G_1^2\sigma_2^2 + G_2^2\sigma_1^2).$$

Multiplying the second inequality by $G_2^2 L_1^2\eta_t/(2\gamma_{t+1})$ and adding it to the first inequality, we have

$$\mathbb{E}\left[F(\mathbf{w}_{t+1}) + \frac{\eta_t G_2^2 L_1^2}{2\gamma_{t+1}}\|\mathbf{u}_{t+1} - g(\mathbf{w}_{t+1})\|_2^2\right] \leq \mathbb{E}\left[F(\mathbf{w}_t) - \frac{\eta_t}{2}\|\nabla F(\mathbf{w}_t)\|_2^2 - \frac{\eta_t}{4}\|\mathcal{M}_t\|_2^2\right]$$
$$+ \frac{\eta_t G_2^2 L_1^2}{2\gamma_{t+1}}\mathbb{E}[\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2] + \frac{\eta_t G_2^2 L_1^2}{2\gamma_{t+1}}\frac{4\eta_t^2 G_2^2}{\gamma_{t+1}}\mathbb{E}[\|\mathcal{M}_t\|_2^2]$$
$$+ \eta_t^2 L_F(G_2^2\sigma_1^2 + G_1^2\sigma_2^2) + \frac{\eta_t G_2^2 L_1^2}{2\gamma_{t+1}}\gamma_{t+1}^2\sigma_0^2 + \frac{\eta_t G_2^2 L_1^2}{2\gamma_{t+1}}\frac{3\eta_t^2 G_2^2}{2}(G_1^2\sigma_2^2 + G_2^2\sigma_1^2).$$

Since $\frac{\eta_t G_2^2 L_1^2}{2\gamma_{t+1}}\frac{4\eta_t^2 G_2^2}{\gamma_{t+1}} \leq \frac{\eta_t}{4}$ due to $\eta_t \leq \frac{\gamma_{t+1}}{\sqrt{8}G_2^2 L_1}$, the term involving $\|\mathcal{M}_t\|_2^2$ will be less than zero. If $\frac{\eta_t}{\gamma_{t+1}} \leq \frac{\eta_{t-1}}{\gamma_t}$, we obtain

$$\mathbb{E}\left[F(\mathbf{w}_{t+1}) + \frac{\eta_t G_2^2 L_1^2}{2\gamma_{t+1}}\|\mathbf{u}_{t+1} - g(\mathbf{w}_{t+1})\|_2^2\right]$$
$$\leq \mathbb{E}\left[F(\mathbf{w}_t) + \frac{\eta_{t-1} G_2^2 L_1^2}{2\gamma_t}\mathbb{E}[\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2\right] - \frac{\eta_t}{2}\mathbb{E}[\|\nabla F(\mathbf{w}_t)\|_2^2] + \frac{\eta_t G_2^2 L_1^2}{2\gamma_{t+1}}\gamma_{t+1}^2\sigma_0^2$$
$$+ \eta_t^2 L_F(G_2^2\sigma_1^2 + G_1^2\sigma_2^2) + \frac{\eta_t G_2^2 L_1^2}{2\gamma_{t+1}}\frac{3\eta_t^2 G_2^2}{2}(G_1^2\sigma_2^2 + G_2^2\sigma_1^2).$$

Applying $\eta_t \leq \frac{\gamma_{t+1}}{\sqrt{8}G_2^2 L_1}$ to the R.H.S, we have

$$\mathbb{E}\left[F(\mathbf{w}_{t+1}) + \frac{\eta_t G_2^2 L_1^2}{2\gamma_{t+1}}\|\mathbf{u}_{t+1} - g(\mathbf{w}_{t+1})\|_2^2\right]$$
$$\leq \mathbb{E}\left[F(\mathbf{w}_t) + \frac{\eta_{t-1} G_2^2 L_1^2}{2\gamma_t}\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2\right] - \frac{\eta_t}{2}\mathbb{E}[\|\nabla F(\mathbf{w}_t)\|_2^2]$$
$$+ \frac{L_1}{2\sqrt{8}}\gamma_{t+1}^2\sigma_0^2 + \eta_t^2(L_F + L_1 G_2^2)(G_2^2\sigma_1^2 + G_1^2\sigma_2^2).$$

Define $\Upsilon_t = F(\mathbf{w}_t) + \frac{\eta_{t-1} G_2^2 L_1^2}{2\gamma_t}\mathbb{E}[\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2$. Then we have $\sum_{t=1}^T(\Upsilon_t - \Upsilon_{t+1}) \leq C_\Upsilon := \Upsilon_1 - F_*$ and

$$\mathbb{E}\left[\sum_{t=1}^{T}\frac{\eta_t}{\sum_{t=1}^{T}\eta_t}\|\nabla F(\mathbf{w}_t)\|_2^2\right] \leq \frac{2C_\Upsilon}{\sum_{t=1}^{T}\eta_t} + \frac{\sum_{t=1}^{T}L_1\gamma_{t+1}^2\sigma_0^2}{\sqrt{8}\sum_{t=1}^{T}\eta_t} + \frac{\sum_{t=1}^{T}2\eta_t^2(L_F+L_1G_2^2)(G_2^2\sigma_1^2+G_1^2\sigma_2^2)}{\sum_{t=1}^{T}\eta_t}.$$

Plugging the values of η_t, γ_t will finish the proof. □

4.2.3 A Straightforward Approach with a Large Batch Size

Before ending this section, we compare the complexity of SCGD with a straightforward approach that uses a large batch size for estimating the gradient. In particular, we update the model parameter by the following:

$$\bar{\mathbf{u}}_t = \frac{1}{B}\sum_{j=1}^{B} g(\mathbf{w}_t;\zeta_{j,t}), \quad \bar{\mathbf{v}}_t = \frac{1}{B}\sum_{i=1}^{B}\nabla g(\mathbf{w}_t;\zeta'_{i,t})\nabla f\left(\bar{\mathbf{u}}_t;\xi_{i,t}\right) \tag{4.14}$$

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t\bar{\mathbf{v}}_t. \tag{4.15}$$

Then under Assumptions 4.1, 4.2, we have

$$\begin{aligned}&\mathbb{E}[\|\bar{\mathbf{v}}_t - \nabla F(\mathbf{w}_t)\|_2^2]\\ &\leq \mathbb{E}\left[\left\|\frac{1}{B}\sum_{i=1}^{B}\nabla g(\mathbf{w}_t;\zeta'_{i,t})\nabla f\left(\bar{\mathbf{u}}_t;\xi_{i,t}\right) - \nabla g(\mathbf{w}_t)\nabla f\left(\bar{\mathbf{u}}_t\right)\right.\right.\\ &\qquad \left.\left. + \nabla g(\mathbf{w}_t)\nabla f\left(\bar{\mathbf{u}}_t\right) - \nabla F(\mathbf{w}_t)\right\|_2^2\right].\end{aligned}$$

Since

$$\begin{aligned}&\mathbb{E}\left[\left\|\frac{1}{B}\sum_{i=1}^{B}\nabla g(\mathbf{w}_t;\zeta'_{i,t})\nabla f\left(\bar{\mathbf{u}}_t;\xi_{i,t}\right) - \nabla g(\mathbf{w}_t)\nabla f\left(\bar{\mathbf{u}}_t\right)\right\|_2\right]\\ &\leq \frac{G_2^2\sigma_1^2+G_1^2\sigma_2^2}{B},\\ &\mathbb{E}\left[\left\|\nabla g(\mathbf{w}_t)\nabla f\left(\bar{\mathbf{u}}_t\right) - \nabla F(\mathbf{w}_t)\right\|_2^2\right] \leq \mathbb{E}[G_2^2L_1^2\|\bar{\mathbf{u}}_t - g(\mathbf{w}_t)\|_2^2] \leq \frac{G_2^2L_1^2\sigma_0^2}{B},\end{aligned}$$

then, $\mathbb{E}[\|\bar{\mathbf{v}}_t - \nabla F(\mathbf{w}_t)\|_2^2] \leq O\left(\frac{L_1^2\sigma_0^2}{B} + \frac{\sigma_1^2+\sigma_2^2}{B}\right)$. Hence, if Assumption 4.4 holds and by setting $B = O(\max(L_1^2\sigma_0^2/\epsilon^2, (\sigma_1^2+\sigma_2^2)/\epsilon^2))$, $\eta = O(1/L_F)$ and $T = O(L_F/\epsilon^2)$, Lemma 4.9 will indicate that the naive approach can find an ϵ-stationary solution. Overall, it yields a sample complexity of

Algorithm 10 SCMA

Input: learning rate schedules $\{\eta_t\}_{t=1}^T$, $\{\gamma_t\}_{t=1}^T$; starting points $\mathbf{w}_0$, $\mathbf{u}_0$, $\mathbf{v}_0$
Let $\mathbf{w}_1 = \mathbf{w}_0 - \eta_0 \mathbf{v}_0$
for $t = 1, \ldots, T$ **do**
  Sample ζ_t, ζ_t' and ξ_t
  Compute the inner function value estimator $\mathbf{u}_t = (1-\gamma_t)\mathbf{u}_{t-1} + \gamma_t g(\mathbf{w}_t; \zeta_t)$
  Compute the vanilla gradient estimator $\mathbf{z}_t = \nabla g(\mathbf{w}_t; \zeta_t')\nabla f(\mathbf{u}_t; \xi_t)$
  Update the MA gradient estimator $\mathbf{v}_t = (1-\beta_t)\mathbf{v}_{t-1} + \beta_t \mathbf{z}_t$
  Update the model $\mathbf{w}$ by $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{v}_t$
end for

$$BT = O\left(\max\left(\frac{L_F L_1^2 \sigma_0^2}{\epsilon^4}, \frac{L_F(\sigma_1^2 + \sigma_2^2)}{\epsilon^4}\right)\right).$$

Critical: Compared with Theorem 4.1, the sample complexity of this naïve approach is improved by an order of magnitude. In comparison to Theorem 4.2, while the order of ϵ remains identical, the dependence on the Lipschitz constant L_1 is reduced. Specifically, SCGD exhibits a dependence of $O(L_1^4)$, whereas the large mini-batch approach achieves $O(L_1^3)$, assuming $L_F = O(L_1)$.

4.3 Stochastic Compositional Momentum Metholds

In this section, we present a method that matches the sample complexity of the large mini-batch approach without using large mini-batches under the smoothness conditions of f and F. The idea is to design a gradient estimator such that its error can be reduced gradually. It turns out this technique, related to the momentum metholds for standard stochastic optimization, is more widely applicable to other problems discussed later in this chapter. Furthermore, we introduce advanced metholds to further improve the complexity to $O(1/\epsilon^3)$ under stronger conditions.

It is worth noting that the results in this section apply to the standard stochastic optimization problem (3.1) under the smoothness assumption of $g(\mathbf{w})$ by setting $f_i(g) = g$ and $L_1 = 0$ in the complexity results and removing the $\mathbf{u}$ update in the algorithm.

4.3.1 Moving-Average Gradient Estimator

The first method is to use the following moving-average gradient estimator:

$$\mathbf{v}_t = (1-\beta_t)\mathbf{v}_{t-1} + \beta_t \nabla g(\mathbf{w}_t; \zeta_t')\nabla f(\mathbf{u}_t; \xi_t), \tag{4.16}$$

where $0 \leq \beta_t < 1$. With $\mathbf{v}_t$, the model parameter is updated by:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{v}_t. \tag{4.17}$$

We present the full steps in Algorithm 10 and refer to it as SCMA.

To understand this method, we can view $\mathbf{v}_t$ as a better estimator of the gradient, with its estimation error gradually decreasing over iterations—a property we will prove shortly. This yields an enhanced stability of momentum-based methods observed in practice.

Connection with Stochastic Momentum Methods

This method is analogous to applying the stochastic momentum method to the ERM problem, using the term $\nabla g(\mathbf{w}_t; \zeta_t')\nabla f(\mathbf{u}_t; \xi_t)$ as a surrogate for the true stochastic gradient. This connection is revealed by reformulating the update into a canonical momentum form:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t' \nabla g(\mathbf{w}_t; \zeta_t')\nabla f(\mathbf{u}_t; \xi_t) + \beta_t'(\mathbf{w}_t - \mathbf{w}_{t-1}), \tag{4.18}$$

where the effective step size and momentum parameters are $\eta_t' = \eta_t\beta_t$ and $\beta_t' = \eta_t(1-\beta_t)/\eta_{t-1}$, respectively. The term $\beta_t'(\mathbf{w}_t - \mathbf{w}_{t-1})$ is the momentum term.

In the special case where f is the identity function, the update is identical to the classical stochastic momentum method (also known as stochastic heavy-ball method), renowned for its accelerated performance on quadratic functions relative to plain gradient descent. Hence, the convergence analysis presented below also applies to the stochastic momentum method for ERM by setting $L_1 = 0$.

Convergence Analysis

First, we prove a generic lemma that establishes the error recursion of $\mathbf{v}_t$.

Lemma 4.7 *Let* $\mathbf{v}_t = (1-\beta_t)\mathbf{v}_{t-1} + \beta_t \mathbf{z}_t$, *where* $\mathbb{E}_t[\mathbf{z}_t] = \mathcal{M}_t$. *If* $\mathbb{E}_t[\|\mathbf{z}_t - \mathcal{M}_t\|_2^2] \leq \sigma^2$, *then we have*

$$\begin{aligned}\mathbb{E}_t\left[\|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|_2^2\right] \leq (1-\beta_t)\|\mathbf{v}_{t-1} - \nabla F(\mathbf{w}_{t-1})\|_2^2 + \beta_t^2\sigma^2 \\ + \frac{2L_F^2}{\beta_t}\|\mathbf{w}_{t-1} - \mathbf{w}_t\|_2^2 + 4\beta_t\|\mathcal{M}_t - \nabla F(\mathbf{w}_t)\|_2^2.\end{aligned} \tag{4.19}$$

Proof. Due to the update formula $\mathbf{v}_t = (1-\beta_t)\mathbf{v}_{t-1} + \beta_t\mathbf{z}_t$, we have

$$
\begin{aligned}
&\mathbb{E}_t\left[\|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|_2^2\right] \\
&= \mathbb{E}_t\left[\|(1-\beta_t)\mathbf{v}_{t-1} + \beta_t \mathbf{z}_t - \nabla F(\mathbf{w}_t)\|_2^2\right] \\
&= \mathbb{E}_t\Bigg[\| \underbrace{(1-\beta_t)\mathbf{v}_{t-1} - \nabla F(\mathbf{w}_t) + \beta_t \mathcal{M}_t}_{\mathbf{a}_t} + \underbrace{\beta_t(\mathbf{z}_t - \mathcal{M}_t)}_{\mathbf{b}_t} \|_2^2\Bigg].
\end{aligned}
$$

Note that $\mathbb{E}_t[\mathbf{a}_t^\top \mathbf{b}_t] = 0$. Besides, we have $\mathbb{E}_t[\|\mathbf{b}_t\|_2^2] \leq \beta_t^2\sigma^2$. Due to Young's inequality, we have $\|a+b\|_2^2 \leq (1+\alpha)\|a\|_2^2 + (1+1/\alpha)\|b\|_2^2$ for any $\alpha > 0$. Hence,

$$
\begin{aligned}
\|\mathbf{a}_t\|_2^2 &= \|(1-\beta_t)(\mathbf{v}_{t-1} - \nabla F(\mathbf{w}_{t-1})) + (1-\beta_t)(\nabla F(\mathbf{w}_{t-1}) - \nabla F(\mathbf{w}_t)) \\
&\quad + \beta_t(\mathcal{M}_t - \nabla F(\mathbf{w}_t))\|_2^2 \\
&\leq (1-\beta_t)^2(1+\beta_t)\|(\mathbf{v}_{t-1} - \nabla F(\mathbf{w}_{t-1}))\|_2^2 \\
&+ (1+\frac{1}{\beta_t})\|(1-\beta_t)(\nabla F(\mathbf{w}_{t-1}) - \nabla F(\mathbf{w}_t)) + \beta_t(\mathcal{M}_t - \nabla F(\mathbf{w}_t))\|_2^2 \\
&\leq (1-\beta_t)\,\|\mathbf{v}_{t-1} - \nabla F(\mathbf{w}_{t-1})\|_2^2 + \frac{2(1+\beta_t)(1-\beta_t)^2}{\beta_t}\,\|\nabla F(\mathbf{w}_{t-1}) - \nabla F(\mathbf{w}_t)\|_2^2 \\
&+ \frac{2(1+\beta_t)\beta_t^2}{\beta_t}\,\|\mathcal{M}_t - \nabla F(\mathbf{w}_t)\|_2^2 \\
&\leq (1-\beta_t)\,\|\mathbf{v}_{t-1} - \nabla F(\mathbf{w}_{t-1})\|_2^2 + \frac{2L_F^2}{\beta_t}\,\|\mathbf{w}_{t-1} - \mathbf{w}_t\|_2^2 + 4\beta_t\,\|\mathcal{M}_t - \nabla F(\mathbf{w}_t)\|_2^2\,.
\end{aligned}
$$

Combining the above results, we finish the proof. □

With the above lemma, we are able to establish the error recursion of $\mathbf{v}_t$ of SCMA.

Lemma 4.8 *Under Assumptions 4.1, 4.2, 4.3, and 4.4, for $t \geq 1$ SCMA satisfies that*

$$
\begin{aligned}
\mathbb{E}_{\xi_t,\zeta_t'}\left[\|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|_2^2\right] &\leq (1-\beta_t)\,\|\mathbf{v}_{t-1} - \nabla F(\mathbf{w}_{t-1})\|_2^2 \\
&+ \frac{2L_F^2}{\beta_t}\,\|\mathbf{w}_{t-1} - \mathbf{w}_t\|_2^2 + 4G_2^2L_1^2\beta_t\,\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2 + \beta_t^2\sigma^2,
\end{aligned} \tag{4.20}
$$

where $\sigma^2 = G_1^2\sigma_2^2 + G_2^2\sigma_1^2$.

Why it matters

The above lemma establishes the recursion of the error of stochastic gradient estimator $\mathbf{v}_t$. It is the key to show that the average of the estimator error of $\mathbf{v}_t$ will converge to zero.

Proof. We denote by $\mathbb{E}_t[\cdot] = \mathbb{E}_{\xi_t,\zeta_t'}[\cdot]$. Let $\mathbf{z}_t = \nabla g(\mathbf{w}_t;\zeta_t')\nabla f(\mathbf{u}_t;\xi_t)$ and $\mathcal{M}_t = \mathbb{E}_t[\mathbf{z}_t] = \nabla g(\mathbf{w}_t)\nabla f(\mathbf{u}_t)$. Lemma 4.4 proves that

$$
\mathbb{E}_t[\|\mathbf{z}_t - \mathcal{M}_t\|_2^2] \leq G_2^2\sigma_1^2 + G_1^2\sigma_2^2, \tag{4.21}
$$

and

$$\begin{aligned}\|\mathcal{M}_t - \nabla F(\mathbf{w}_t)\|_2^2 &= \|\nabla g(\mathbf{w}_t)\nabla f(\mathbf{u}_t) - \nabla g(\mathbf{w}_t)\nabla f(g(\mathbf{w}_t))\|_2^2 \\ &\leq G_2^2 L_1^2 \|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2.\end{aligned}$$

Plugging these two results into Lemma 4.7, we finish the proof. □

Critical: If we use the same random sample ζ_t to compute

$$\mathbf{z}_t = \nabla g(\mathbf{w}_t; \zeta_t)\nabla f(\mathbf{u}_t; \xi_t),$$

then $\mathcal{M}_t = \mathbb{E}_{\xi_t,\zeta_t}[\mathbf{z}_t]$ is not equal to $\nabla g(\mathbf{w}_t)\nabla f(\mathbf{u}_t)$. However, we just need to assume that $\mathbb{E}_{\zeta_t,\xi_t}[\|\mathbf{z}_t - \mathcal{M}_t\|_2^2]$ is bounded and $\|\nabla g(\mathbf{w}_t; \zeta_t)\|_2^2 \leq G_2$. Then

$$\begin{aligned}\|\mathcal{M}_t - \nabla F(\mathbf{w}_t)\|_2^2 &= \|\mathbb{E}_{\zeta_t}\nabla g(\mathbf{w}_t; \zeta_t)\nabla f(\mathbf{u}_t) - \mathbb{E}_{\zeta_t}\nabla g(\mathbf{w}_t; \zeta_t)\nabla f(g(\mathbf{w}_t))\|_2^2 \\ &\leq \mathbb{E}_{\zeta_t}\|\nabla g(\mathbf{w}_t; \zeta_t)\nabla f(\mathbf{u}_t) - \nabla g(\mathbf{w}_t; \zeta_t)\nabla f(g(\mathbf{w}_t))\|_2^2 \\ &\leq \mathbb{E}_{\zeta_t}[\|\nabla g(\mathbf{w}_t; \zeta_t)\|_2^2\|\nabla f(\mathbf{u}_t) - \nabla f(g(\mathbf{w}_t))\|_2^2] \\ &\leq \mathbb{E}_{\zeta_t}\left[G_2^2 L_1^2 \|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2\right].\end{aligned}$$

The following analysis will proceed in the same manner.

To enjoy the above recursion of the gradient estimator's error, we state the following lemma, which is a variant of the standard descent lemma of gradient descent.

Lemma 4.9 *For the update* $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{v}_t, t \geq 0$, *if* $\eta_t \leq 1/(2L_F)$, *we have*

$$F(\mathbf{w}_{t+1}) \leq F(\mathbf{w}_t) + \frac{\eta_t}{2}\|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\|_2^2 - \frac{\eta_t}{2}\|\nabla F(\mathbf{w}_t)\|_2^2 - \frac{1}{4\eta_t}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2. \tag{4.22}$$

Why it matters

This lemma ensures that if the stochastic gradient error satisfies $\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^T \|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\|_2^2\right] \to 0$, then the convergence of $\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^T \|\nabla F(\mathbf{w}_t)\|_2^2\right]$ to zero is guaranteed.

Proof. Due to the smoothness of F, we have

$$
\begin{aligned}
&F(\mathbf{w}_{t+1}) \leq F(\mathbf{w}_t) + \nabla F(\mathbf{w}_t)^\top (\mathbf{w}_{t+1} - \mathbf{w}_t) + \frac{L_F}{2} \|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 \\
&= F(\mathbf{w}_t) + (\nabla F(\mathbf{w}_t) - \mathbf{v}_t)^\top (\mathbf{w}_{t+1} - \mathbf{w}_t) + \mathbf{v}_t^\top (\mathbf{w}_{t+1} - \mathbf{w}_t) + \frac{L_F}{2} \|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 \\
&= F(\mathbf{w}_t) - \eta_t (\nabla F(\mathbf{w}_t) - \mathbf{v}_t)^\top \mathbf{v}_t - \left(\frac{1}{\eta_t} - \frac{L_F}{2}\right) \|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 \\
&= F(\mathbf{w}_t) + \eta_t \|(\nabla F(\mathbf{w}_t) - \mathbf{v}_t)\|_2^2 - \eta_t (\nabla F(\mathbf{w}_t) - \mathbf{v}_t)^\top \nabla F(\mathbf{w}_t) \\
&- \left(\frac{1}{\eta_t} - \frac{L_F}{2}\right) \|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 .
\end{aligned}
$$

Since $(\nabla F(\mathbf{w}_t) - \mathbf{v}_t)^\top \nabla F(\mathbf{w}_t) = \frac{1}{2}\left(\|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\|_2^2 + \|\nabla F(\mathbf{w}_t)\|_2^2 - \|\mathbf{v}_t\|_2^2\right)$, then we have

$$
\begin{aligned}
&F(\mathbf{w}_{t+1}) \leq F(\mathbf{w}_t) + \eta_t \|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\|_2^2 - \left(\frac{1}{\eta_t} - \frac{L_F}{2}\right) \|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 \\
&- \frac{\eta_t}{2}\left(\|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\|_2^2 + \|\nabla F(\mathbf{w}_t)\|_2^2 - \|\mathbf{v}_t\|_2^2\right) \\
&= F(\mathbf{w}_t) + \frac{\eta_t}{2} \|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\|_2^2 - \frac{\eta_t}{2} \|\nabla F(\mathbf{w}_t)\|_2^2 - \left(\frac{1}{2\eta_t} - \frac{L_F}{2}\right) \|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 .
\end{aligned}
$$

□

To prove the final convergence of SCMA, we present a useful lemma.

Lemma 4.10 *If $\eta_t \leq 1/L$, assume that there exist non-negative sequences $A_t, B_t, \Gamma_t, \Delta_t, \delta_t, t \geq 0$ satisfying:*

$$
\begin{aligned}
&(*) A_{t+1} \leq A_t + \eta_t \Delta_t - \eta_t B_t - \eta_t \Gamma_t \\
&(\sharp) \Delta_{t+1} \leq (1 - \beta_{t+1}) \Delta_t + C_1 \beta_{t+1} \delta_{t+1} + \frac{C_2 \eta_t^2}{\beta_{t+1}} \Gamma_t + \beta_{t+1}^2 \sigma^2, \\
&(\diamond) \delta_{t+1} \leq (1 - \gamma_{t+1}) \delta_t + \frac{C_3 \eta_t^2}{\gamma_{t+1}} \Gamma_t + \gamma_{t+1}^2 \sigma'^2.
\end{aligned}
$$

Let $\Upsilon_t = A_t + \frac{\eta_{t-1}}{\beta_t} \Delta_t + \frac{C_1 \eta_{t-1}}{\gamma_t} \delta_t$. If $\frac{\eta_t}{\beta_{t+1}} \leq \frac{\eta_{t-1}}{\beta_t}$, $\frac{\eta_t}{\gamma_{t+1}} \leq \frac{\eta_{t-1}}{\gamma_t}$, $\eta_t \leq \min(\frac{\beta_{t+1}}{\sqrt{4C_2}}, \frac{\gamma_{t+1}}{\sqrt{8C_1C_3}})$, and $\Upsilon_t \geq A_*$, then we have

$$
\sum_{t=0}^{T-1} \frac{1}{\sum_{t=0}^{T-1} \eta_t} (\eta_t B_t + \frac{1}{2} \eta_t \Gamma_t) \leq \frac{C_\Upsilon}{\sum_{t=0}^{T-1} \eta_t} + \frac{\sum_{t=0}^{T-1} \left(\eta_t \beta_{t+1} \sigma^2 + 2 C_1 \eta_t \gamma_{t+1} \sigma'^2\right)}{\sum_{t=0}^{T-1} \eta_t},
$$

where $C_\Upsilon = \Upsilon_0 - A_* \leq A_0 - A_* + \frac{1}{2\sqrt{C_2}} \Delta_0 + \sqrt{\frac{C_1}{8C_3}} \delta_0$.

If $\beta = \frac{\epsilon^2}{3\sigma^2}, \gamma = \frac{\epsilon^2}{6C_1 \sigma'^2}, \eta = \min(\frac{1}{L}, \frac{\beta}{\sqrt{4C_2}}, \frac{\gamma}{\sqrt{8C_1C_3}})$, then in order to guarantee

$$\sum_{t=0}^{T-1} \frac{1}{T}(B_t + \frac{1}{2}\Gamma_t) \leq \epsilon^2.$$

the iteration complexity is the in the order of

$$T = O\left(\max\left\{\frac{C_\Upsilon L}{\epsilon^2}, \frac{C_\Upsilon \sigma^2\sqrt{C_2}}{\epsilon^4}, \frac{C_\Upsilon\sqrt{C_1C_3}C_1\sigma'^2}{\epsilon^4}\right\}\right).$$

Critical: If $(*)$, $(\sharp)$, $(\diamond)$ hold in expectation, then the concluding inequalities also hold in expectation.

Proof. The proof is constructive. The idea is to construct a telescoping series of $A_t + a_t\Delta_t + b_t\delta_t$ with some appropriate sequences of a_t, b_t. First, we have

$$\begin{aligned}
&A_{t+1} + a_{t+1}\Delta_{t+1} + b_{t+1}\delta_{t+1} \leq A_t + \eta_t\Delta_t - \eta_t B_t - \eta_t\Gamma_t \\
&+ a_{t+1}(1-\beta_{t+1})\Delta_t + a_{t+1}C_1\beta_{t+1}\delta_{t+1} + a_{t+1}\frac{C_2\eta_t^2}{\beta_{t+1}}\Gamma_t + a_{t+1}\beta_{t+1}^2\sigma^2 \\
&+ b_{t+1}(1-\gamma_{t+1})\delta_t + b_{t+1}\frac{C_3\eta_t^2}{\gamma_{t+1}}\Gamma_t + b_{t+1}\gamma_{t+1}^2\sigma'^2.
\end{aligned}$$

Let $a_{t+1} = \eta_t/\beta_{t+1} \leq \eta_{t-1}/\beta_t$ and $b_{t+1} = C_1\eta_t(1+\gamma_{t+1})/\gamma_{t+1}$, we have

$$\begin{aligned}
&A_{t+1} + \frac{\eta_t}{\beta_{t+1}}\Delta_{t+1} + (C_1\eta_t\frac{1+\gamma_{t+1}}{\gamma_{t+1}} - C_1\eta_t)\delta_{t+1} \leq A_t - \eta_t B_t - \eta_t\Gamma_t \\
&+ \left(\eta_t + \frac{\eta_t}{\beta_{t+1}}(1-\beta_{t+1})\right)\Delta_t + \frac{C_2\eta_t^3}{\beta_{t+1}^2}\Gamma_t + \eta_t\beta_{t+1}\sigma^2 \\
&+ C_1\eta_t\frac{1+\gamma_{t+1}}{\gamma_{t+1}}(1-\gamma_{t+1})\delta_t + \frac{C_3C_1\eta_t^3(1+\gamma_{t+1})}{\gamma_{t+1}^2}\Gamma_t + C_1\eta_t(1+\gamma_{t+1})\gamma_{t+1}\sigma'^2.
\end{aligned}$$

Thus,

$$\begin{aligned}
&A_{t+1} + \frac{\eta_t}{\beta_{t+1}}\Delta_{t+1} + \frac{C_1\eta_t}{\gamma_{t+1}}\delta_{t+1} \leq A_t + \frac{\eta_t}{\beta_{t+1}}\Delta_t + \frac{C_1\eta_t}{\gamma_{t+1}}\delta_t \\
&- \eta_t B_t - \left(\eta_t - \frac{C_2\eta_t^3}{\beta_{t+1}^2} - \frac{C_3C_1\eta_t^3(1+\gamma_{t+1})}{\gamma_{t+1}^2}\right)\Gamma_t \\
&+ \eta_t\beta_{t+1}\sigma^2 + C_1\eta_t(1+\gamma_{t+1})\gamma_{t+1}\sigma'^2.
\end{aligned}$$

Since $\eta_t/\beta_{t+1} \leq \eta_{t-1}/\beta_t$ and $\eta_t/\gamma_{t+1} \leq \eta_{t-1}/\gamma_t$ and $\gamma_{t+1} \leq 1$, we have

$$
\begin{aligned}
&A_{t+1} + \frac{\eta_t}{\beta_{t+1}}\Delta_{t+1} + \frac{C_1\eta_t}{\gamma_{t+1}}\delta_{t+1} \le A_t + \frac{\eta_{t-1}}{\beta_t}\Delta_t + \frac{C_1\eta_{t-1}}{\gamma_t}\delta_t \\
&- \eta_t B_t - \left(\eta_t - \frac{C_2\eta_t^3}{\beta_{t+1}^2} - \frac{2C_3C_1\eta_t^3}{\gamma_{t+1}^2}\right)\Gamma_t \\
&+ \eta_t\beta_{t+1}\sigma^2 + 2C_1\eta_t\gamma_{t+1}\sigma'^2.
\end{aligned}
$$

Since $C_2\eta_t^3/\beta_{t+1}^2 \le \eta_t/4$ (because $\eta_t \le \beta_{t+1}/\sqrt{4C_2}$) and $2C_3C_1\eta_t^3/\gamma_{t+1}^2 \le \eta_t/4$ (because $\eta_t \le \gamma_{t+1}/\sqrt{8C_1C_3}$), we have

$$
\begin{aligned}
&A_{t+1} + \frac{\eta_t}{\beta_{t+1}}\Delta_{t+1} + \frac{C_1\eta_t}{\gamma_{t+1}}\delta_{t+1} \le A_t + \frac{\eta_{t-1}}{\beta_t}\Delta_t + \frac{C_1\eta_{t-1}}{\gamma_t}\delta_t \\
&- \eta_t B_t - \frac{1}{2}\eta_t\Gamma_t + \eta_t\beta_{t+1}\sigma^2 + 2C_1\eta_t\gamma_{t+1}\sigma'^2.
\end{aligned}
$$

Define $\Upsilon_{t+1} = A_{t+1} + \frac{\eta_t}{\beta_{t+1}}\Delta_{t+1} + \frac{C_1\eta_t}{\gamma_{t+1}}\delta_{t+1}$, we have

$$
\eta_t B_t + \frac{1}{2}\eta_t\Gamma_t \le \Upsilon_t - \Upsilon_{t+1} + \eta_t\beta_{t+1}\sigma^2 + 2C_1\eta_t\gamma_{t+1}\sigma'^2.
$$

Hence

$$
\sum_{t=0}^{T-1}(\eta_t B_t + \frac{1}{2}\eta_t\Gamma_t) \le \Upsilon_0 - A_* + \sum_{t=0}^{T-1}\left(\eta_t\beta_{t+1}\sigma^2 + 2C_1\eta_t\gamma_{t+1}\sigma'^2\right).
$$

Next, let us consider $\eta_t = \eta, \beta_t = \beta, \gamma_t = \gamma$. Then we have

$$
\sum_{t=0}^{T-1}\frac{1}{T}(B_t + \frac{1}{2}\Gamma_t) \le \frac{C_\Upsilon}{T} + \left(\beta\sigma^2 + 2C_1\gamma\sigma'^2\right).
$$

In order to ensure the RHS is less than ϵ^2, it suffices to have

$$
\beta = \frac{\epsilon^2}{3\sigma^2}, \quad \gamma = \frac{\epsilon^2}{6C_1\sigma'^2}, \quad T = \frac{C_\Upsilon}{3\epsilon^2\eta}.
$$

Since

$$
\eta = \min\left(\frac{1}{L}, \frac{\beta}{\sqrt{4C_2}}, \frac{\gamma}{\sqrt{8C_1C_3}}\right),
$$

thus the order of T becomes

$$
\begin{aligned}
T &= O\left(\max\left\{\frac{C_\Upsilon L}{\epsilon^2}, \frac{C_\Upsilon\sqrt{C_2}}{\epsilon^2\beta}, \frac{C_\Upsilon\sqrt{C_1C_3}}{\gamma\epsilon^2}\right\}\right) \\
&= O\left(\max\left\{\frac{C_\Upsilon L_F}{\epsilon^2}, \frac{C_\Upsilon\sigma^2\sqrt{C_2}}{\epsilon^4}, \frac{C_\Upsilon\sqrt{C_1C_3}C_1\sigma'^2}{\epsilon^4}\right\}\right),
\end{aligned}
$$

where

$$C_\Upsilon = A_0 - A_* + \frac{\eta}{\beta}\Delta_0 + \frac{C_1\eta}{\gamma}\delta_0 \leq A_0 - A_* + \frac{1}{2\sqrt{C_2}}\Delta_0 + \frac{\sqrt{C_1}}{\sqrt{8C_3}}\delta_0.$$

□

Finally, let us prove the convergence of SCMA.

Theorem 4.3 *Suppose Assumptions 4.1, 4.2, 4.3, and 4.4 hold. For the SCMA algorithm, set the parameters as follows:* $\beta = \frac{\epsilon^2}{3\sigma^2}$, $\gamma = \frac{\epsilon^2}{6C_1\sigma_0^2}$, *and* $\eta = \min\left(\frac{1}{2L_F}, \frac{\beta}{\sqrt{4C_2}}, \frac{\gamma}{\sqrt{8C_1C_3}}\right)$, *where* $\sigma^2 = G_2^2\sigma_1^2 + G_1^2\sigma_2^2$, $C_1 = 4G_2^2L_1^2$, $C_2 = 4L_F^2$, $C_3 = 2G_2^2$. *Then, the following*

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1}\left\{\frac{1}{4}\|\mathbf{v}_t\|_2^2 + \|\nabla F(\mathbf{w}_t)\|_2^2\right\}\right] \leq \epsilon^2$$

holds, with an iteration complexity of

$$T = O\left(\max\left\{\frac{C_\Upsilon L_F}{\epsilon^2}, \frac{C_\Upsilon\sigma^2 L_F}{\epsilon^4}, \frac{C_\Upsilon L_1^3\sigma_0^2}{\epsilon^4}\right\}\right).$$

where $C_\Upsilon := 2\left(F(\mathbf{w}_0) - F_*\right) + \frac{1}{8L_F}\|\nabla F(\mathbf{w}_0) - \mathbf{v}_0\|_2^2 + \frac{L_1}{2}\|\mathbf{u}_0 - g(\mathbf{w}_0)\|_2^2$.

Why it matters

Insights 1: Theorem 4.3 indicates that SCMA enjoys the same complexity of $O(1/\epsilon^4)$ for finding an ϵ-stationary solution as SGD for ERM. In addition, the averaged estimation error of the moving-average gradient estimator $\mathbf{v}_t$, i.e., $\mathbb{E}[\frac{1}{T}\sum_{t=0}^{T-1}\|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|_2^2]$, converges to zero as $T \to \infty$.
Insights 2: We can apply the above result to the Momentum method (6.2) for solving the standard stochastic optimization $\min_{\mathbf{w}} F(\mathbf{w}) := \mathbb{E}_\zeta[g(\mathbf{w};\zeta)]$ by setting $L_1 = 0$. The complexity of the Momentum method is

$$T = O\left(\max\left\{\frac{(F(\mathbf{w}_0) - F_*)L_F}{\epsilon^2}, \frac{(F(\mathbf{w}_0) - F_*)\sigma^2 L_F}{\epsilon^4}, \frac{\|\nabla F(\mathbf{w}_0) - \mathbf{v}_0\|_2^2\,\sigma^2}{\epsilon^4}\right\}\right),$$

which is no worse than that of SGD in Theorem 3.3. The key advantage of the Momentum method over SGD is that it ensures the averaged estimation error of the moving-average gradient estimator $\mathbf{v}_t$ converge to zero.
The convergence bound also suggests that it is better to initialize $\mathbf{v}_0$ in a way such that $\|\nabla F(\mathbf{w}_0) - \mathbf{v}_0\|_2^2$ is small, e.g., using the mini-batch gradient at $\mathbf{w}_0$ instead of initializing it to zero.

Proof. The three inequalities in Lemma 4.8, 4.9 and 4.1 that we have proved so far are

$$(*)F(\mathbf{w}_{t+1}) \leq F(\mathbf{w}_t) + \frac{\eta_t}{2}\left\|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\right\|_2^2 - \frac{\eta_t}{2}\left\|\nabla F(\mathbf{w}_t)\right\|_2^2 - \frac{\eta_t}{4}\left\|\mathbf{v}_t\right\|_2^2, t \geq 0$$

$$(\sharp)\mathbb{E}\left[\left\|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\right\|_2^2\right] \leq \mathbb{E}[(1-\beta_t)\left\|\mathbf{v}_{t-1} - \nabla F(\mathbf{w}_{t-1})\right\|_2^2]$$
$$+ \mathbb{E}\left[4G_2^2L_1^2\beta_t\left\|\mathbf{u}_t - g(\mathbf{w}_t)\right\|_2^2 + \frac{2L_F^2\eta_{t-1}^2}{\beta_t}\left\|\mathbf{v}_{t-1}\right\|_2^2 + \beta_t^2\sigma^2\right],$$

$$(\diamond)\mathbb{E}\left[\left\|\mathbf{u}_t - g(\mathbf{w}_t)\right\|_2^2\right] \leq \mathbb{E}[(1-\gamma_t)\left\|\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1})\right\|_2^2]$$
$$+ \mathbb{E}\left[\frac{G_2^2\eta_{t-1}^2}{\gamma_t}\left\|\mathbf{v}_{t-1}\right\|_2^2 + \gamma_t^2\sigma_0^2\right].$$

Define $A_t = 2(F(\mathbf{w}_t) - F_*)$ and $B_t = \left\|\nabla F(\mathbf{w}_t)\right\|_2^2$, $\Gamma_t = \left\|\mathbf{v}_t\right\|_2^2/2$, $\Delta_t = \left\|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\right\|_2^2$, $\delta_t = \left\|\mathbf{u}_t - g(\mathbf{w}_t)\right\|_2^2$, and $\Upsilon_t = A_t + \frac{\eta_{t-1}}{\beta_t}\Delta_t + \frac{C_1\eta_{t-1}}{\gamma_t}\delta_t$.

Then the three inequalities satisfy that in Lemma 4.10 with $C_1 = 4G_2^2L_1^2, C_2 = 4L_F^2, C_3 = 2G_2^2, \sigma^2 = G_1^2\sigma_2^2 + G_2^2\sigma_1^2, \sigma'^2 = \sigma_0^2$. Then $\eta_t, \beta_t, \gamma_t$ satisfy

$$\frac{\eta_t}{\beta_{t+1}} \leq \frac{\eta_{t-1}}{\beta_t}, \frac{\eta_t}{\gamma_{t+1}} \leq \frac{\eta_{t-1}}{\gamma_t}, \eta_t \leq \min(\frac{\beta_{t+1}}{\sqrt{4C_2}}, \frac{\gamma_{t+1}}{\sqrt{8C_1C_3}}).$$

Then we have

$$\mathbb{E}\left[\sum_{t=0}^{T-1}\frac{1}{\sum_{t=0}^{T-1}\eta_t}(\eta_t\|\nabla F(\mathbf{w}_t)\|_2^2 + \frac{\eta_t}{4}\left\|\mathbf{v}_t\right\|_2^2)\right]$$
$$\leq \frac{C_\Upsilon}{\sum_{t=1}\eta_t} + \frac{\sum_{t=0}^{T-1}\left(\eta_t\beta_{t+1}\sigma^2 + 2C_1\eta_t\gamma_{t+1}\sigma_0^2\right)}{\sum_{t=0}^{T-1}\eta_t}.$$

Since the setting of η, γ, β satisfy that in Lemma 4.10, the order of T becomes

$$T = O\left(\max\left\{\frac{C_\Upsilon L_F}{\epsilon^2}, \frac{C_\Upsilon\sigma^2\sqrt{C_2}}{\epsilon^4}, \frac{C_\Upsilon\sqrt{C_1C_3}C_1\sigma_0^2}{\epsilon^4}\right\}\right)$$
$$= O\left(\max\left\{\frac{C_\Upsilon L_F}{\epsilon^2}, \frac{C_\Upsilon\sigma^2 L_F}{\epsilon^4}, \frac{C_\Upsilon L_1^3\sigma_0^2}{\epsilon^4}\right\}\right),$$

where

$$C_\Upsilon = 2(F(\mathbf{w}_0) - F_*) + \frac{1}{2\sqrt{C_2}}\|\mathbf{v}_0 - \nabla F(\mathbf{w}_0)\|_2^2 + \frac{\sqrt{C_1}}{\sqrt{8C_3}}\|\mathbf{u}_0 - g(\mathbf{w}_0)\|_2^2$$
$$= 2(F(\mathbf{w}_0) - F_*) + \frac{1}{4L_F}\left\|\mathbf{v}_0 - \nabla F(\mathbf{w}_0)\right\|_2^2 + \frac{L_1}{2}\left\|\mathbf{u}_0 - g(\mathbf{w}_0)\right\|_2^2.$$

□

4.3.2 STORM Estimators

We can further reduce the error of the gradient estimator by using advanced variance reduction techniques under stronger assumptions. We make the following assumptions.

Assumption 4.6. *There exists* $L_1, G_1 > 0$ *such that*

(i) $\mathbb{E}[\|\nabla f(g;\xi) - \nabla f(g';\zeta)\|_2^2] \leq L_1^2\|g - g'\|_2^2, \forall g, g';$
(ii) $\mathbb{E}[\|\nabla f(g;\xi)\|_2^2] \leq G_1^2, \forall g.$

Assumption 4.7. *There exists* $L_2, G_2 > 0$ *such that*

(i) $\mathbb{E}[\|\nabla g(\mathbf{w};\zeta) - \nabla g(\mathbf{w}';\zeta)\|_2^2] \leq L_2^2\|\mathbf{w} - \mathbf{w}'\|_2^2, \forall \mathbf{w}, \mathbf{w}';$
(ii) $\mathbb{E}[\|\nabla g(\mathbf{w};\zeta)\|_2^2] \leq G_2^2, \forall \mathbf{w}.$

Due to Jensen's inequality, Assumption (4.6)(i) implies the Lipschitz continuity assumption of ∇f in Assumption (4.1)(i). Similarly, Assumption (4.7)(i) implies that in Assumption 4.2(i), respectively. Hence, Assumption (4.6)(i) and Assumption (4.7)(i) are stronger, which are referred to as mean-square smoothness condition of f and g.

The STORM estimator

Let us first discuss a generic STORM estimator, an improved variant of the moving average estimator. Without loss of generality, we consider estimating a sequence of mappings $\{\mathcal{M}(\mathbf{w}_t)\}_{t=1}^T$ through their stochastic values at each iteration $\{\mathcal{M}(\mathbf{w}_t;\zeta_t)\}_{t=1}^T$, where $\mathbb{E}_{\zeta_t}[\mathcal{M}(\mathbf{w}_t;\zeta_t)] = \mathcal{M}(\mathbf{w}_t) \in \mathbb{R}^{d'}$. We assume the mapping $\mathcal{M}$ satisfies:

$$\mathbb{E}_{\zeta}[\|\mathcal{M}(\mathbf{w};\zeta) - \mathcal{M}(\mathbf{w}';\zeta)\|_2^2] \leq G^2\|\mathbf{w} - \mathbf{w}'\|_2^2, \forall \mathbf{w}, \mathbf{w}';$$

The STORM estimator is give by a sequence of $\mathcal{U}_1, \ldots, \mathcal{U}_T$, where

$$\mathcal{U}_t = (1-\gamma_t)\mathcal{U}_{t-1} + \gamma_t\mathcal{M}(\mathbf{w}_t;\zeta_t) + (1-\gamma_t)(\mathcal{M}(\mathbf{w}_t;\zeta_t) - \mathcal{M}(\mathbf{w}_{t-1};\zeta_t)), \tag{4.23}$$

and $\gamma_t \in (0,1)$.
It augments the moving-average estimator by adding an extra term $(1-\gamma_t)(\mathcal{M}(\mathbf{w}_t;\zeta_t) - \mathcal{M}(\mathbf{w}_{t-1};\zeta_t))$, which can be viewed as an error correction term.

Applying the STORM estimator to estimating the sequence of $\{g(\mathbf{w}_t)\}_{t\geq 1}$, we have the following sequence:

$$\mathbf{u}_t = (1-\gamma_t)\mathbf{u}_{t-1} + \gamma_t g(\mathbf{w}_t;\zeta_t) + (1-\gamma_t)(g(\mathbf{w}_t;\zeta_t) - g(\mathbf{w}_{t-1};\zeta_t)). \tag{4.24}$$

Given $\mathbf{u}_t$, we can compute a moving-average gradient estimator (4.16) similar to SCMA. However, this will not yield an improved rate compared with SCMA. To

Algorithm 11 SCST

Input: learning rate schedules $\{\eta_t\}_{t=0}^T$, $\{\gamma_t\}_{t=1}^T$; starting points $\mathbf{w}_0$, $\mathbf{u}_0$, $\mathbf{v}_0$
Let $\mathbf{w}_1 = \mathbf{w}_0 - \eta_0 \mathbf{v}_0$
for $t = 1, \ldots, T$ **do**
  Sample ζ_t, ζ_t' and ξ_t
  Update the inner function value estimator

$$\mathbf{u}_t = (1-\gamma_t)\mathbf{u}_{t-1} + \gamma_t g(\mathbf{w}_t;\zeta_t) + (1-\gamma_t)(g(\mathbf{w}_t;\zeta_t) - g(\mathbf{w}_{t-1};\zeta_t))$$

  Compute the vanilla gradient estimator $\mathbf{z}_t = \nabla g(\mathbf{w}_t;\zeta_t')\nabla f(\mathbf{u}_t;\xi_t)$
  Compute $\tilde{\mathbf{z}}_{t-1} = \nabla g(\mathbf{w}_{t-1};\zeta_t')\nabla f(\mathbf{u}_{t-1};\xi_t)$
  Update the STORM gradient estimator $\mathbf{v}_t = (1-\beta_t)\mathbf{v}_{t-1} + \beta_t \mathbf{z}_t + (1-\beta_t)(\mathbf{z}_t - \tilde{\mathbf{z}}_{t-1})$
  Update the model by $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{v}_t$
end for

reduce the estimator error of the gradient, we apply another STORM estimator to estimate $\mathcal{M}_t = \nabla g(\mathbf{w}_t)\nabla f(\mathbf{u}_t)$. This is computed by the following sequence:

$$\begin{aligned}\mathbf{v}_t &= (1-\beta_t)\mathbf{v}_{t-1} + \beta_t \nabla g(\mathbf{w}_t;\zeta_t')\nabla f(\mathbf{u}_t;\xi_t) \\ &\quad + (1-\beta_t)(\nabla g(\mathbf{w}_t;\zeta_t')\nabla f(\mathbf{u}_t;\xi_t) - \nabla g(\mathbf{w}_{t-1};\zeta_t')\nabla f(\mathbf{u}_{t-1};\xi_t)).\end{aligned} \tag{4.25}$$

With $\mathbf{v}_t$, we update the model parameters by

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \mathbf{v}_t.$$

The full steps of this method is presented in Algorithm 11, which is referred to as SCST.

Connection with Variance-reduced metholds for Non-convex optimization

In the special case where f is the identity function, the update is identical to the classical variance-reduced method (also known as STROM) for non-convex optimization $\min_{\mathbf{w}} \mathbb{E}_\zeta [g(\mathbf{w};\zeta)]$, i.e.,

$$\begin{aligned}&\mathbf{v}_t = (1-\beta_t)\mathbf{v}_{t-1} + \beta_t \nabla g(\mathbf{w}_t;\zeta_t') + (1-\beta_t)(\nabla g(\mathbf{w}_t;\zeta_t') - \nabla g(\mathbf{w}_{t-1};\zeta_t')), \\ &\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{v}_t.\end{aligned} \tag{4.26}$$

It is renowned for its improved complexity of $O(1/\epsilon^3)$ better than the complexity $O(1/\epsilon^4)$ of SGD for finding an ϵ-stationary solution.

Convergence Analysis

We first prove a general result of the STORM estimator that applies to both $\mathbf{u}_t$ and $\mathbf{v}_t$.

Lemma 4.11 *Consider* $\mathbf{v}_t = (1-\beta_t)\mathbf{v}_{t-1} + \beta_t \mathbf{z}_t + (1-\beta_t)(\mathbf{z}_t - \tilde{\mathbf{z}}_{t-1})$, *where* $\beta_t \in (0,1)$. *Let* $\mathbb{E}_t$ *denote the expectation over randomness associated with* $\mathbf{z}_t, \tilde{\mathbf{z}}_{t-1}$ *condition on the randomness before t-the iteration. If* $\mathbb{E}_t[\mathbf{z}_t] = \mathcal{M}_t$ *and* $\mathbb{E}_t[\tilde{\mathbf{z}}_{t-1}] = \mathcal{M}_{t-1}$. *If* $\mathbb{E}_t[\|\mathbf{z}_t - \mathcal{M}_t\|_2^2] \leq \sigma^2$, *then we have*

$$\mathbb{E}_t\left[\|\mathbf{v}_t - \mathcal{M}_t\|_2^2\right] \leq (1-\beta_t)\|\mathbf{v}_{t-1} - \mathcal{M}_{t-1}\|_2^2 + \mathbb{E}_t[2\|\mathbf{z}_t - \tilde{\mathbf{z}}_{t-1}\|_2^2] + 2\beta_t^2\sigma^2.$$

Proof.

$$\begin{aligned}
&\mathbb{E}_t\left[\|\mathbf{v}_t - \mathcal{M}_t\|_2^2\right] \\
&= \mathbb{E}_t\left[\|(1-\beta_t)\mathbf{v}_{t-1} - \mathcal{M}_t + \beta_t\mathbf{z}_t + (1-\beta_t)(\mathbf{z}_t - \tilde{\mathbf{z}}_{t-1})\|_2^2\right] \\
&= \mathbb{E}_t\left[\|(1-\beta_t)(\mathbf{v}_{t-1} - \mathcal{M}_{t-1}) + (1-\beta_t)((\mathbf{z}_t - \tilde{\mathbf{z}}_{t-1}) - (\mathcal{M}_t - \mathcal{M}_{t-1})) \right.\\
&\qquad \left. + \beta_t(\mathbf{z}_t - \mathcal{M}_t)\|_2^2\right].
\end{aligned}$$

Note that

$$\begin{aligned}
&\mathbb{E}_t\left[\langle(1-\beta_t)(\mathbf{v}_{t-1} - \mathcal{M}_{t-1}),\right. \\
&\quad \left.(1-\beta_t)((\mathbf{z}_t - \tilde{\mathbf{z}}_{t-1}) - (\mathcal{M}_t - \mathcal{M}_{t-1})) + \beta_t(\mathbf{z}_t - \mathcal{M}_t)\rangle\right] = 0.
\end{aligned}$$

Then,

$$\begin{aligned}
&\mathbb{E}_t\left[\|\mathbf{v}_t - \mathcal{M}_t\|_2^2\right] \leq (1-\beta_t)^2\|\mathbf{v}_t - \mathcal{M}_{t-1}\|_2^2 \\
&+ \|(1-\beta_t)((\mathbf{z}_t - \tilde{\mathbf{z}}_{t-1}) - (\mathcal{M}_t - \mathcal{M}_{t-1})) + \beta_t(\mathbf{z}_t - \mathcal{M}_t)\|_2^2 \\
&\overset{(\diamond)}{\leq} (1-\beta_t)^2\|\mathbf{v}_t - \mathcal{M}_{t-1}\|_2^2 \\
&+ 2(1-\beta_t)^2\mathbb{E}_t[\|((\mathbf{z}_t - \tilde{\mathbf{z}}_{t-1}) - (\mathcal{M}_t - \mathcal{M}_{t-1}))\|_2^2] + 2\beta_t^2\mathbb{E}_t[\|\mathbf{z}_t - \mathcal{M}_t\|_2^2] \\
&\overset{(*)}{\leq} (1-\beta_t)^2\|\mathbf{v}_{t-1} - \mathcal{M}_{t-1}\|_2^2 + 2(1-\beta_t)^2\mathbb{E}_t[\|\mathbf{z}_t - \tilde{\mathbf{z}}_{t-1}\|_2^2] + 2\beta_t^2\sigma^2,
\end{aligned}$$

where $(\diamond)$ uses the Young's inequality, $(*)$ uses the fact that $\mathbb{E}[\|a - \mathbb{E}[a]\|_2^2] \leq \mathbb{E}[\|a\|_2^2]$, and $\mathbb{E}_t[\mathbf{z}_t - \tilde{\mathbf{z}}_{t-1}] = \mathcal{M}_t - \mathcal{M}_{t-1}$. □

Let us first prove an error recursion of $\mathbf{u}_t$ in the lemma below.

Lemma 4.12 *Under Assumption (4.7)(ii), we have:*

$$\begin{aligned}
&\mathbb{E}_{\zeta_t}\left[\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2\right] \leq (1-\gamma_t)\|\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1})\|_2^2 + 2\gamma_t^2\sigma_0^2 + 2G_2^2\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2 \\
&\mathbb{E}_{\zeta_t}\left[\|\mathbf{u}_t - \mathbf{u}_{t-1}\|_2^2\right] \leq 2\gamma_t^2\sigma_0^2 + 4\gamma_t^2\|\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1})\|_2^2 + 6G_2^2\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2.
\end{aligned}$$

Why it matters

Compared to the error recursion of $\mathbf{u}_t$ to that in Lemma 4.1, the improvement comes from the last term reducing from $\frac{2G_2^2\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2}{\gamma_t}$ to $2G_2^2\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2$.

Proof. The first part follows directly from Lemma 4.11 by noting the mean-Lipschitz continuity of $g(\mathbf{w};\zeta)$. To prove the second part, we proceed as follows:

$$\begin{aligned}
&\mathbb{E}_t\left[\|\mathbf{u}_t-\mathbf{u}_{t-1}\|_2^2\right]\\
=&\mathbb{E}_t\left[\|\gamma_t\left(g(\mathbf{w}_t;\zeta_t)-\mathbf{u}_{t-1}\right)+(1-\gamma_t)\left(g(\mathbf{w}_t;\zeta_t)-g(\mathbf{w}_{t-1};\zeta_t)\right)\|_2^2\right]\\
\le&\mathbb{E}_t\left[2\gamma_t^2\|(g(\mathbf{w}_t;\zeta_t)-\mathbf{u}_{t-1})\|_2^2+2(1-\gamma_t)^2\|g(\mathbf{w}_t;\zeta_t)-g(\mathbf{w}_{t-1};\zeta_t)\|_2^2\right]\\
\le&\mathbb{E}_t\left[2\gamma_t^2\|(g(\mathbf{w}_t;\zeta_t)-\mathbf{u}_{t-1})\|_2^2\right]+2(1-\gamma_t)^2G_2^2\|\mathbf{w}_t-\mathbf{w}_{t-1}\|_2^2.
\end{aligned}$$

Next, we bound the first term on the RHS as

$$\begin{aligned}
&\mathbb{E}_t\left[2\gamma_t^2\|(g(\mathbf{w}_t;\zeta_t)-\mathbf{u}_{t-1})\|_2^2\right]=\mathbb{E}_t\left[2\gamma_t^2\|(g(\mathbf{w}_t;\zeta_t)-g(\mathbf{w}_t)+g(\mathbf{w}_t)-\mathbf{u}_{t-1})\|_2^2\right]\\
&\le 2\gamma_t^2\sigma_0^2+2\gamma_t^2\|g(\mathbf{w}_t)-\mathbf{u}_{t-1}\|_2^2\\
&\le 2\gamma_t^2\sigma_0^2+2\gamma_t^2\|g(\mathbf{w}_t)-g(\mathbf{w}_{t-1})+g(\mathbf{w}_{t-1})-\mathbf{u}_{t-1}\|_2^2\\
&\le 2\gamma_t^2\sigma_0^2+4\gamma_t^2\|g(\mathbf{w}_{t-1})-\mathbf{u}_{t-1}\|_2^2+4\gamma_t^2G_2^2\|\mathbf{w}_t-\mathbf{w}_{t-1}\|_2^2,
\end{aligned}$$

where the first inequality uses the fact $\mathbb{E}\left[g(\mathbf{w}_t;\zeta_t)-g(\mathbf{w}_t)\right]=0$. Combining the above results, we finish the proof. □

Next, we build an error recursion of $\|\mathbf{v}_t-\mathcal{M}_t\|_2^2$.

Lemma 4.13 *Let $\sigma^2=G_2^2\sigma_1^2+G_1^2\sigma_2^2$. Under Assumptions (4.6) and Assumption (4.7), (4.25) satisfies that*

$$\begin{aligned}
&\mathbb{E}_{\zeta_t',\xi_t}\left[\|\mathbf{v}_t-\mathcal{M}_t\|_2^2\right]\le(1-\beta_t)\|\mathbf{v}_{t-1}-\mathcal{M}_{t-1}\|_2^2 \qquad (4.27)\\
&+16G_2^2L_1^2\gamma_t^2\|\mathbf{u}_{t-1}-g(\mathbf{w}_{t-1})\|_2^2+(24G_2^4L_1^2+4G_1^2L_2^2)\|\mathbf{w}_t-\mathbf{w}_{t-1}\|_2^2\\
&+2\beta_t^2\sigma^2+8G_2^2L_1^2\gamma_t^2\sigma_0^2.
\end{aligned}$$

Proof. First, (4.21) gives $\mathbb{E}_t[\|\mathbf{z}_t-\mathcal{M}_t\|_2^2]\le\sigma^2$. Second,

$$\begin{aligned}
&\mathbb{E}_t\left[\|\mathbf{z}_t-\tilde{\mathbf{z}}_{t-1}\|_2^2\right]\\
&=\mathbb{E}_t[\|\nabla g(\mathbf{w}_t;\zeta_t')\nabla f(\mathbf{u}_t;\xi_t)-\nabla g(\mathbf{w}_{t-1};\zeta_t')\nabla f(\mathbf{u}_{t-1};\xi_t)\|_2^2]\\
&=\mathbb{E}_t[\|\nabla g(\mathbf{w}_t;\zeta_t')\nabla f(\mathbf{u}_t;\xi_t)-\nabla g(\mathbf{w}_t;\zeta_t')\nabla f(\mathbf{u}_{t-1};\xi_t)\\
&\qquad+\nabla g(\mathbf{w}_t;\zeta_t')\nabla f(\mathbf{u}_{t-1};\xi_t)-\nabla g(\mathbf{w}_{t-1};\zeta_t')\nabla f(\mathbf{u}_{t-1};\xi_t)\|_2^2]\\
&\overset{(\triangle)}{\le}2G_2^2L_1^2\|\mathbf{u}_t-\mathbf{u}_{t-1}\|_2^2+2G_1^2L_2^2\|\mathbf{w}_t-\mathbf{w}_{t-1}\|_2^2,
\end{aligned}$$

where $(\triangle)$ uses the Assumption (4.6)(i) and Assumption (4.7)(i). It then follows:

$$\begin{aligned}
\mathbb{E}_t\left[\|\mathbf{v}_t-\mathcal{M}_t\|_2^2\right]\le&(1-\beta_t)^2\|\mathbf{v}_{t-1}-\mathcal{M}_{t-1}\|_2^2\\
&+4G_2^2L_1^2\|\mathbf{u}_t-\mathbf{u}_{t-1}\|_2^2+4G_1^2L_2^2\|\mathbf{w}_t-\mathbf{w}_{t-1}\|_2^2+2\beta_t^2\sigma^2.
\end{aligned}$$

By using the second inequality of Lemma 4.12, i.e.,

$$\mathbb{E}_{\zeta_t}\left[\|\mathbf{u}_t-\mathbf{u}_{t-1}\|_2^2\right]\leq 2\gamma_t^2\sigma_0^2+4\gamma_t^2\|\mathbf{u}_{t-1}-g(\mathbf{w}_{t-1})\|_2^2+6G_2^2\|\mathbf{w}_t-\mathbf{w}_{t-1}\|_2^2,$$

we have

$$\begin{aligned}\mathbb{E}_t\left[\|\mathbf{v}_t-\mathcal{M}_t\|_2^2\right]&\leq(1-\beta_t)\|\mathbf{v}_{t-1}-\mathcal{M}_{t-1}\|_2^2+16G_2^2L_1^2\gamma_t^2\|\mathbf{u}_{t-1}-g(\mathbf{w}_{t-1})\|_2^2\\&+(24G_2^4L_1^2+4G_1^2L_2^2)\|\mathbf{w}_t-\mathbf{w}_{t-1}\|_2^2+2\beta_t^2\sigma^2+8G_2^2L_1^2\gamma_t^2\sigma_0^2.\end{aligned}$$

□

Similar to Lemma 4.9, we have the following descent lemma.

Lemma 4.14 *For the update* $\mathbf{w}_{t+1}=\mathbf{w}_t-\eta_t\mathbf{v}_t, t\geq 0$, *if* $\eta_t\leq 1/(2L_F)$ *we have*

$$\begin{aligned}F(\mathbf{w}_{t+1})\leq F(\mathbf{w}_t)&+\eta_tG_2^2L_1^2\|\mathbf{u}_t-g(\mathbf{w}_t)\|_2^2+\eta_t\|\mathbf{v}_t-H_t\|_2^2\\&-\frac{\eta_t}{2}\|\nabla F(\mathbf{w}_t)\|_2^2-\frac{1}{4\eta_t}\|\mathbf{w}_{t+1}-\mathbf{w}_t\|_2^2.\end{aligned}\tag{4.28}$$

This lemma can be proved following that of lemma 4.9 by bound $\|\mathbf{v}_t-\nabla F(\mathbf{w}_t)\|_2^2\leq 2\|\mathbf{v}_t-\mathcal{M}_t\|_2^2+2\|\mathcal{M}_t-\nabla F(\mathbf{w}_t)\|_2^2\leq 2\|\mathbf{v}_t-\mathcal{M}_t\|_2^2+2G_2^2L_1^2\|\mathbf{u}_t-g(\mathbf{w}_t)\|_2^2$.

Lemma 4.15 *For* $\eta_t\leq 1/L$, *the non-negative sequences* $A_t,B_t,\Gamma_t,\Delta_t,\delta_t,t\geq 0$ *satisfy:*

$$\begin{aligned}&(*)A_{t+1}\leq A_t+\eta_t\Delta_t+\eta_t\delta_t-\eta_tB_t-\eta_t\Gamma_t\\&(\sharp)\Delta_{t+1}\leq(1-\beta_{t+1})\Delta_t+C_1\gamma_{t+1}^2\delta_t+C_2\eta_t^2\Gamma_t+\beta_{t+1}^2\sigma^2+\gamma_{t+1}^2\sigma'^2,\\&(\diamond)\delta_{t+1}\leq(1-\gamma_{t+1})\delta_t+C_3\eta_t^2\Gamma_t+\gamma_{t+1}^2\sigma''^2.\end{aligned}$$

Let $\Upsilon_{t+1}=A_{t+1}+\frac{c}{\eta_t}\Delta_{t+1}+\frac{c'}{\eta_t}\delta_{t+1}\geq A_*$. *Suppose* $c,c',\eta_t,\gamma_t,\beta_t$ *satisfy:*

$$\begin{aligned}&C_2c+C_3c'\leq\frac{1}{2},\quad \eta_t+\frac{c}{\eta_t}(1-\beta_{t+1})\leq\frac{c}{\eta_{t-1}},\\&\eta_t+\frac{c}{\eta_t}C_1\gamma_{t+1}^2+\frac{c'}{\eta_t}(1-\gamma_{t+1})\leq\frac{c'}{\eta_{t-1}}.\end{aligned}\tag{4.29}$$

Then,

$$\sum_{t=0}^{T-1}(\eta_tB_t+\frac{1}{2}\eta_t\Gamma_t)\leq C_\Upsilon+\sum_{t=0}^{T-1}\left(\frac{c\beta_{t+1}^2}{\eta_t}\sigma^2+\frac{c\gamma_{t+1}^2}{\eta_t}\sigma'^2+\frac{c'\gamma_{t+1}^2}{\eta_t}\sigma''^2\right).\tag{4.30}$$

If we set $c=\frac{1}{4C_2},c'=\frac{1}{4C_3},\beta_t=\frac{\epsilon\eta\sqrt{C_2}}{\sigma}$, $\gamma_t=\min\left(\frac{\epsilon\eta\sqrt{C_2}}{\sigma'},\frac{\epsilon\eta\sqrt{C_3}}{\sigma''},\frac{C_2}{2C_3C_1}\right)$, *and* $\eta_t=\eta=\min\left(\frac{1}{L},\frac{\epsilon}{4\sqrt{C_2}\sigma},\frac{\epsilon\sqrt{C_2}}{8C_3\sigma'},\frac{\epsilon}{8\sqrt{C_3}\sigma''},\frac{\sqrt{C_2}}{4C_3\sqrt{C_1}}\right)$, *then in order to grantee*

$$\sum_{t=0}^{T-1} \frac{1}{T}(B_t + \frac{1}{2}\Gamma_t) \leq \epsilon^2, \tag{4.31}$$

the iteration complexity is in the order of

$$T = O\left(\max\left\{\frac{C_\Upsilon L}{\epsilon^2}, \frac{C_\Upsilon C_3\sqrt{C_1/C_2}}{\epsilon^2}, \frac{C_\Upsilon\sigma\sqrt{C_2}}{\epsilon^3}, \frac{C_\Upsilon C_3\sigma'}{\epsilon^3\sqrt{C_2}}, \frac{C_\Upsilon\sigma''\sqrt{C_3}}{\epsilon^3}\right\}\right)$$

where $C_\Upsilon = \Upsilon_0 - A_* = A_0 + \frac{1}{4C_2\eta}\Delta_0 + \frac{1}{4C_3\eta}\delta_0 - A_*$.

Critical: If $(*)$, $(\sharp)$, $(\diamond)$ hold in expectation, then the two inequalities in (4.30) and (4.31) hold in expectation.

Proof. The proof is constructive. The idea is to multiply the second inequality by a_{t+1} and the third inequality by b_{t+1} such that we can construct a telescoping series of $A_t + a_t\Delta_t + b_t\delta_t$. First, we have

$$\begin{aligned}
&A_{t+1} + a_{t+1}\Delta_{t+1} + b_{t+1}\delta_{t+1} \leq A_t + \eta_t\Delta_t + \eta_t\delta_t - \eta_t B_t - \eta_t\Gamma_t \\
&+ a_{t+1}(1-\beta_{t+1})\Delta_t + a_{t+1}C_1\gamma_{t+1}^2\delta_t + a_{t+1}C_2\eta_t^2\Gamma_t + a_{t+1}\beta_{t+1}^2\sigma^2 + a_{t+1}\gamma_{t+1}^2\sigma'^2. \\
&+ b_{t+1}(1-\gamma_{t+1})\delta_t + b_{t+1}C_3\eta_t^2\Gamma_t + b_{t+1}\gamma_{t+1}^2\sigma''^2.
\end{aligned}$$

Let $a_{t+1} = c/\eta_t$ and $b_{t+1} = c'/\eta_t$, we have

$$\begin{aligned}
&A_{t+1} + \frac{c}{\eta_t}\Delta_{t+1} + \frac{c'}{\eta_t}\delta_{t+1} \leq A_t - \eta_t B_t - \eta_t\Gamma_t \\
&+ \left(\eta_t + \frac{c}{\eta_t}(1-\beta_{t+1})\right)\Delta_t + C_2 c\eta_t\Gamma_t + \frac{c\beta_{t+1}^2}{\eta_t}\sigma^2 + \frac{c\gamma_{t+1}^2}{\eta_t}\sigma'^2 \\
&+ \left(\eta_t + \frac{c}{\eta_t}C_1\gamma_{t+1}^2 + \frac{c'}{\eta_t}(1-\gamma_{t+1})\right)\delta_t + C_3 c'\eta_t\Gamma_t + \frac{c'\gamma_{t+1}^2}{\eta_t}\sigma''^2.
\end{aligned}$$

With (4.29) we have

$$\begin{aligned}
&A_{t+1} + \frac{c}{\eta_t}\Delta_{t+1} + \frac{c'}{\eta_t}\delta_{t+1} \leq A_t + \frac{c}{\eta_{t-1}}\Delta_t + \frac{c'}{\eta_{t-1}}\delta_t - \eta_t B_t - \frac{1}{2}\eta_t\Gamma_t \\
&+ \frac{c\beta_{t+1}^2}{\eta_t}\sigma^2 + \frac{c\gamma_{t+1}^2}{\eta_t}\sigma'^2 + \frac{c'\gamma_{t+1}^2}{\eta_t}\sigma''^2
\end{aligned}$$

Define $\Upsilon_{t+1} = A_{t+1} + \frac{c}{\eta_t}\Delta_{t+1} + \frac{c'}{\eta_t}\delta_{t+1}$, we have

$$\eta_t B_t + \frac{1}{2}\eta_t\Gamma_t \leq \Upsilon_t - \Upsilon_{t+1} + \frac{c\beta_{t+1}^2}{\eta_t}\sigma^2 + \frac{c\gamma_{t+1}^2}{\eta_t}\sigma'^2 + \frac{c'\gamma_{t+1}^2}{\eta_t}\sigma''^2.$$

Hence

$$\sum_{t=0}^{T-1}(\eta_t B_t + \frac{1}{2}\eta_t\Gamma_t) \leq \Upsilon_0 - A_* + \sum_{t=0}^{T-1}\left(\frac{c\beta_{t+1}^2}{\eta_t}\sigma^2 + \frac{c\gamma_{t+1}^2}{\eta_t}\sigma'^2 + \frac{c'\gamma_{t+1}^2}{\eta_t}\sigma''^2\right).$$

Next, let us consider $\eta_t = \eta, \beta_t = \beta, \gamma_t = \gamma$. Then we have

$$\sum_{t=0}^{T-1}\frac{1}{T}(B_t + \frac{1}{2}\Gamma_t) \leq \frac{\Upsilon_0 - A_*}{\eta T} + \left(\frac{c\beta^2}{\eta^2}\sigma^2 + \frac{c\gamma^2}{\eta^2}\sigma'^2 + \frac{c'\gamma^2}{\eta^2}\sigma''^2\right).$$

In order to ensure the RHS is less than ϵ^2, it suffices to have

$$\beta = \frac{\epsilon\eta}{2\sqrt{c}\sigma}, \quad \gamma = \min\left(\frac{\epsilon\eta}{2\sqrt{c}\sigma'}, \frac{\epsilon\eta}{2\sqrt{c'}\sigma''}\right), \quad T = \frac{C_\Upsilon}{4\epsilon^2\eta}.$$

To ensure (4.29), it suffices to have

$$\eta^2 \leq c\beta, \quad C_1 c\gamma \leq c'/2, \quad \eta^2 \leq c'\gamma/2, \quad c = \frac{1}{4C_2}, \quad c' = \frac{1}{4C_3}.$$

As a result, if we set

$$\begin{aligned}\eta &= \min\left(\frac{1}{L}, \frac{\epsilon\sqrt{c}}{2\sigma}, \frac{\epsilon c'}{4\sqrt{c}\sigma'}, \frac{\epsilon\sqrt{c'}}{4\sigma''}, \frac{c'}{2\sqrt{cC_1}}\right)\\ &= \min\left(\frac{1}{L}, \frac{\epsilon}{4\sqrt{C_2}\sigma}, \frac{\epsilon\sqrt{C_2}}{8C_3\sigma'}, \frac{\epsilon}{8\sqrt{C_3}\sigma''}, \frac{\sqrt{C_2}}{4C_3\sqrt{C_1}}\right)\\ \beta &= \frac{\epsilon\eta\sqrt{C_2}}{\sigma}, \quad \gamma = \min\left(\frac{\epsilon\eta\sqrt{C_2}}{\sigma'}, \frac{\epsilon\eta\sqrt{C_3}}{\sigma''}, \frac{C_2}{2C_3C_1}\right),\end{aligned}$$

we have

$$\sum_{t=0}^{T-1}\frac{1}{T}(B_t + \frac{1}{2}\Gamma_t) \leq \epsilon^2.$$

Plugging the values of η into the requirement of T yields the order of T.

□

Theorem 4.4 *Suppose that Assumptions 4.3, 4.6,and 4.7 hold. For SCST, in order to guarantee*

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1}\left\{\frac{1}{4}\|\mathbf{v}_t\|_2^2 + \|\nabla F(\mathbf{w}_t)\|_2^2\right\}\right] \leq \epsilon^2,$$

we can set the parameters as $\eta = \min\{O(\frac{1}{L_F}), O(\frac{\epsilon}{L_1\sigma}), O(\frac{\epsilon}{L_1^2\sigma_0})\}$, $\beta = O(\frac{\epsilon\eta L_1}{\sigma})$, *and* $\gamma = \min\{O(\frac{\epsilon\eta}{\sigma_0}, O(1)\}$, *and the iteration complexity is*

$$T = O\left(\max(\frac{C_\Upsilon L_1(\sigma_1+\sigma_2)}{\epsilon^3}, \frac{C_\Upsilon\sigma_0 L_1^2}{\epsilon^3}, \frac{C_\Upsilon L_F}{\epsilon^2})\right),$$

where $C_Y = O(F(\mathbf{w}_0) - F_* + \frac{1}{L_1^2\eta}\|\nabla g(\mathbf{w}_0)\nabla f(\mathbf{u}_0) - \mathbf{v}_0\|_2^2 + \frac{1}{L_1^2\eta}\|g(\mathbf{w}_0) - \mathbf{u}_0\|_2^2)$.

Why it matters

We only explicitly maintain the dependence on L_1, which will have implications when we handle non-smooth f in next Chapter.

The above theorem can help us establish an improved iteration complexity of $O(1/\epsilon^3)$. First, we need to ensure $C_Y = O(1)$, which can be satisfied by using a large initial batch size. In particular, we can set $\mathbf{u}_0 = \frac{1}{B_0}\sum_{i=1}^{B_0} g(\mathbf{w}_0;\zeta_i)$, $\mathbf{v}_0 = \frac{1}{B_0}\sum_{i=1}^{B_0}\nabla g(\mathbf{w}_0;\zeta_i')\nabla f(\mathbf{u}_0;\xi_i)$, where $\{\zeta_i,\zeta_i',\zeta_i\}_{i=1}^{B_0}$ are independent random variables. Thus, we have $\mathbb{E}[\|\mathbf{u}_0 - g(\mathbf{w}_0)\|_2^2] \leq O(\frac{1}{B_0})$ and $\mathbb{E}[\|\mathbf{v}_0 - \nabla g(\mathbf{w}_0)\nabla f(\mathbf{u}_0)\|_2^2] \leq O(\frac{1}{B_0})$. Hence, if we set $B_0 = O(\frac{\sigma}{L_1\epsilon}, \frac{\sigma_0}{\epsilon})$ we have $C_Y = O(1)$. This initial batch size requirement can be removed by using a decreasing parameters $\eta_t = O(1/t^{1/3}), \beta_t = O(1/t^{2/3}), \gamma_t = O(1/t^{2/3})$.

Compared to the result of SCMA in Theorem 4.3, SCST has a higher order of step size η and a smaller order of iteration complexity.

Proof. Let us recall the three inequalities in Lemma 4.14, 4.13 and 4.12:

$$(*)\ F(\mathbf{w}_{t+1}) \leq F(\mathbf{w}_t) + \eta_t G_2^2 L_1^2\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2 + \eta_t\|\mathbf{v}_t - \mathcal{M}_t\|_2^2 - \frac{\eta_t}{2}\|\nabla F(\mathbf{w}_t)\|_2^2 - \frac{1}{4\eta_t}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2,$$

$$(\sharp)\ \mathbb{E}\left[\|\mathbf{v}_t - \mathcal{M}_t\|_2^2\right] \leq \mathbb{E}[(1-\beta_t)\|\mathbf{v}_{t-1} - \mathcal{M}_{t-1}\|_2^2] + 16G_2^2L_1^2\gamma_t^2\|\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1})\|_2^2] + \mathbb{E}[(24G_2^4L_1^2 + 4G_1^2L_2^2)\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2 + 2\beta_t^2\sigma^2 + 8G_2^2L_1^2\gamma_t^2\sigma_0^2],$$

$$(\diamond)\ \mathbb{E}_{\zeta_t}\left[\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2\right] \leq (1-\gamma_t)\|\mathbf{u}_{t-1} - g(\mathbf{w}_{t-1})\|_2^2 + \mathbb{E}[2G_2^2\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2 + 2\gamma_t^2\sigma_0^2].$$

Define

$$A_t = F(\mathbf{w}_t) - F_*, \quad B_t = \|\nabla F(\mathbf{w}_t)\|_2^2/2,$$
$$\Gamma_t = \|\mathbf{v}_t\|_2^2/4, \quad \Delta_t = \|\mathbf{v}_t - H_t\|_2^2, \quad \delta_t = L_1^2G_2^2\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2.$$

They satisfy the three inequalities marked by $*, \sharp, \diamond$ in Lemma 4.15 with Then we have $C_1 = 16, C_2 = O(G_2^4L_1^2 + G_1^2L_2^2), C_3 = O(L_1^2G_2^2), \sigma^2 = O(G_2^2\sigma_1^2 + G_1^2\sigma_2^2), \sigma'^2 = O(L_1^2G_2^2\sigma_0^2), \sigma''^2 = O(L_1^2G_2^2\sigma_0^2)$. Plugging these into Lemma 4.15, we can finish the proof. □

4.4 Non-smooth (Non-convex) Regularized Problems

In this section, we consider the following regularized stochastic compositional optimization:

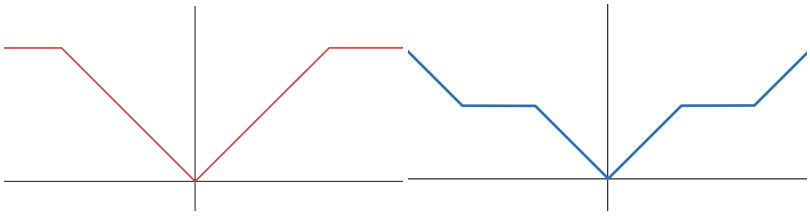


Fig. 4.1: Left: the capped ℓ_1-norm regularizer; Right: a non-convex PAR regularizer

$$\min_{\mathbf{w}\in\mathbb{R}^d} \bar{F}(\mathbf{w}) := \mathbb{E}_\xi f(\mathbb{E}_\zeta[g(\mathbf{w};\zeta)];\xi) + r(\mathbf{w}), \tag{4.32}$$

where r is a non-smooth regularizer, which is potentially non-convex. This includes constrained problems, where $r(\mathbf{w}) = \mathbb{I}_{0-\infty}(\mathbf{w} \in \mathcal{W})$. For example, the KL-constrained DRO (2.19) has a constraint $\lambda \geq 0$.

We extend the definition of ϵ-stationary solution of a smooth function to the non-smooth composite function by noting that $\partial(F + r)(\mathbf{w}) = \nabla F(\mathbf{w}) + \partial r(\mathbf{w})$.

Definition 4.1 (ϵ-stationary solution) A solution $\mathbf{w}$ is called an ϵ-stationary solution to $\min_{\mathbf{w}\in\mathbb{R}^d} F(\mathbf{w}) + r(\mathbf{w})$ where F is smooth and r is non-differentiable, if $\text{dist}(0, \nabla F(\mathbf{w}) + \partial r(\mathbf{w})) \leq \epsilon$.

To handle non-smoothness or r, we assume the proximal mapping of r is simple to compute:

$$\text{prox}_r(\hat{\mathbf{w}}) = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \frac{1}{2}\|\mathbf{w} - \hat{\mathbf{w}}\|_2^2 + r(\mathbf{w}).$$

Below, we give some examples of non-convex regularizers and their proximal mappings, whose derivations are left as exercises for interested readers.

Examples

Example 4.4 (Capped ℓ_1-norm)**.** *It is defined as $r(\mathbf{w}) = \lambda \sum_{i=1}^d \psi(w_i)$, where $\psi(w_i) = \min(|w_i|, \theta)$ (cf. Figure (4.1)). It penalizes small coefficients heavily (encouraging sparsity) but stops penalizing once coefficients are large enough. It was shown to reduce the bias issue of LASSO, which cannot exactly recover the non-zero coefficients under some conditions. Its proximal mapping is given by*

$$prox_{\lambda\psi}(u) = \begin{cases} x_1 = \min(sign(u)(|u| - \lambda)_+, \theta) & \text{if } h(x_1; u) < h(x_2; u) \\ x_2 = \max(|u|, \theta) & \text{otherwise}, \end{cases} \tag{4.33}$$

where $h(x;u) = \frac{1}{2}(x-u)^2 + \lambda \min(|x|, \theta)$. *Similar non-convex sparse regularizers include minimax concave penalty (MCP) and Smoothly Clipped Absolute Deviation (SCAD).*

Example 4.5 (Nonconvex Piecewise Affine Regularization (PAR))**.** *A non-convex PAR is defined as* $r(\mathbf{w}) = \lambda \sum_{i=1}^{d} \psi(w_i)$ *(cf Figure (4.1)), where*

$$\psi(x) = \begin{cases} |x| - kq & \text{if } kq \le |x| \le \frac{2k+1}{2}q, \\ \frac{k+1}{2}q & \text{if } \frac{2k+1}{2}q \le |x| \le (k+1)q, \end{cases} \quad k = 0, 1, \ldots, \tag{4.34}$$

Its proximal mapping is defined as:

- *When the regularization strength* $\lambda \le q$*, we have*

$$prox_{\lambda\psi}(u) = \begin{cases} sign(u)kq & \text{if } kq \le |u| \le kq + \lambda, \\ sign(u)(|u| - \lambda) & \text{if } kq + \lambda \le |u| \le \frac{2k+1}{2}q + \frac{\lambda}{2}, \\ sign(u)|u| & \text{if } \frac{2k+1}{2}q + \frac{\lambda}{2} \le |u| \le (k+1)q. \end{cases} \tag{4.35}$$

- *When the regularization strength* $\lambda \ge q$*, we have*

$$prox_{\lambda\psi}(u) = sign(u) \left\lfloor \frac{|u| - \frac{\lambda}{2}}{q} \right\rceil q. \tag{4.36}$$

where $\lfloor \cdot \rceil$ *denotes the nearest integer. When* λ *exceeds a certain threshold (e.g.,* $\lambda \ge q$*), the proximal operator becomes a* ***hard quantizer****, mapping inputs exactly to discrete levels in a quantization set* $Q = \{0, \pm q, \pm 2q, \pm 3q, \ldots\}$.

Algorithms

We can easily extend SCMA and SCST to solving the non-smooth regularized SCO problems using the following update:

$$\mathbf{w}_{t+1} = \arg\min \frac{1}{2\eta_t} \|\mathbf{w} - (\mathbf{w}_t - \eta_t \mathbf{v}_t)\|_2^2 + r(\mathbf{w}), \tag{4.37}$$

where $\mathbf{v}_t$ is the MA or STORM gradient estimator as in SCMA or SCST.

Convergence Analysis

We first present a lemma similar to Lemma 4.9.

Lemma 4.16 *Consider the update in (4.37), if* $\eta_t \le \frac{1}{4L_F}$ *then we have*

$$\bar{F}(\mathbf{w}_t) - \bar{F}(\mathbf{w}_{t+1}) \leq \eta_t \|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|_2^2 - \frac{\eta_t}{10} dist(0, \partial \bar{F}(\mathbf{w}_{t+1}))^2 - \frac{1}{80\eta_t}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2.$$

Proof. Recall the update of $\mathbf{w}_{t+1}$:

$$\mathbf{w}_{t+1} \in \arg\min_{\mathbf{w}\in\mathbb{R}^d} \left\{ r(\mathbf{w}) + \frac{1}{2\eta_t}\|\mathbf{w} - (\mathbf{w}_t - \eta_t \mathbf{v}_t)\|_2^2 \right\}.$$

Then following variational analysis, we have

$$-\mathbf{v}_t - \frac{1}{\eta_t}(\mathbf{w}_{t+1} - \mathbf{w}_t) \in \partial r(\mathbf{w}_{t+1}),$$

which implies that

$$\nabla F(\mathbf{w}_{t+1}) - \mathbf{v}_t - \frac{1}{\eta_t}(\mathbf{w}_{t+1} - \mathbf{w}_t) \in \nabla F(\mathbf{w}_{t+1}) + \partial r(\mathbf{w}_{t+1}) = \partial \bar{F}(\mathbf{w}_{t+1}). \tag{4.38}$$

Hence, we have

$$\text{dist}(0, \partial \bar{F}(\mathbf{w}_{t+1}))^2 \leq \|\nabla F(\mathbf{w}_{t+1}) - \mathbf{v}_t - \frac{1}{\eta_t}(\mathbf{w}_{t+1} - \mathbf{w}_t)\|_2^2 \tag{4.39}$$

Due to the update of $\mathbf{w}_{t+1}$, we also have

$$r(\mathbf{w}_{t+1}) + \langle \mathbf{v}_t, \mathbf{w}_{t+1} - \mathbf{w}_t \rangle + \frac{1}{2\eta_t}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 \leq r(\mathbf{w}_t). \tag{4.40}$$

Since $F(\mathbf{w})$ is smooth with parameter L_F, then

$$F(\mathbf{w}_{t+1}) \leq F(\mathbf{w}_t) + \langle \nabla F(\mathbf{w}_t), \mathbf{w}_{t+1} - \mathbf{w}_t \rangle + \frac{L_F}{2}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2. \tag{4.41}$$

Combining these two inequalities (4.40) and (4.41) we get

$$\bar{F}(\mathbf{w}_{t+1}) + \langle \mathbf{v}_t - \nabla F(\mathbf{w}_t), \mathbf{w}_{t+1} - \mathbf{w}_t \rangle \leq \bar{F}(\mathbf{w}_t) - (\frac{1}{2\eta_t} - \frac{L_F}{2})\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2.$$

From the above inequality, we obtain two results. The first result is

$$
\begin{aligned}
&\frac{2}{\eta_t}\langle \mathbf{v}_t - \nabla F(\mathbf{w}_{t+1}), \mathbf{w}_{t+1} - \mathbf{w}_t\rangle \\
&\leq \frac{2(\bar{F}(\mathbf{w}_t) - \bar{F}(\mathbf{w}_{t+1}))}{\eta_t} - \frac{1}{\eta_t}(\frac{1}{\eta_t} - L_F)\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 \\
&+ \frac{2}{\eta_t}\langle \nabla F(\mathbf{w}_t) - \nabla F(\mathbf{w}_{t+1}), \mathbf{w}_{t+1} - \mathbf{w}_t\rangle \\
&\leq \frac{2(\bar{F}(\mathbf{w}_t) - \bar{F}(\mathbf{w}_{t+1}))}{\eta_t} - \frac{1}{\eta_t}(\frac{1}{\eta_t} - 3L_F)\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2. \qquad (4.42)
\end{aligned}
$$

The second result is

$$
\begin{aligned}
&(\frac{1}{2\eta_t} - \frac{L_F}{2})\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 \leq \bar{F}(\mathbf{w}_t) - \bar{F}(\mathbf{w}_{t+1}) + \langle \nabla F(\mathbf{w}_t) - \mathbf{v}_t, \mathbf{w}_{t+1} - \mathbf{w}_t\rangle \\
&= \bar{F}(\mathbf{w}_t) - \bar{F}(\mathbf{w}_{t+1}) + \eta_t\|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\|_2^2 + \frac{1}{4\eta_t}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2.
\end{aligned}
$$

If $\frac{L_F}{2} \leq \frac{1}{8\eta_t}$, i.e., $\eta_t \leq \frac{1}{4L_F}$, the above inequality indicates:

$$
\frac{1}{8\eta_t}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 \leq \bar{F}(\mathbf{w}_t) - \bar{F}(\mathbf{w}_{t+1}) + \eta_t\|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\|_2^2. \qquad (4.43)
$$

To proceed, we have

$$
\begin{aligned}
&\|\mathbf{v}_t - \nabla F(\mathbf{w}_{t+1}) + \frac{1}{\eta_t}(\mathbf{w}_{t+1} - \mathbf{w}_t)\|_2^2 \\
&= 2\langle \mathbf{v}_t - \nabla F(\mathbf{w}_{t+1}), \frac{1}{\eta_t}(\mathbf{w}_{t+1} - \mathbf{w}_t)\rangle + \|\mathbf{v}_t - \nabla F(\mathbf{w}_{t+1})\|_2^2 + \frac{1}{\eta_t^2}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2.
\end{aligned}
$$

Adding the above inequality to (4.42) we have

$$
\begin{aligned}
&\|\mathbf{v}_t - \nabla F(\mathbf{w}_{t+1}) + \frac{1}{\eta_t}(\mathbf{w}_{t+1} - \mathbf{w}_t)\|_2^2 \\
&\leq \frac{2(\bar{F}(\mathbf{w}_t) - \bar{F}(\mathbf{w}_{t+1}))}{\eta_t} - \frac{1}{\eta_t}(\frac{1}{\eta_t} - 3L_F)\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 \\
&+ \|\mathbf{v}_t - \nabla F(\mathbf{w}_{t+1})\|_2^2 + \frac{1}{\eta_t^2}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 \\
&= \frac{2(\bar{F}(\mathbf{w}_t) - \bar{F}(\mathbf{w}_{t+1}))}{\eta_t} + \frac{3L_F}{\eta_t}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 + \|\mathbf{v}_t - \nabla F(\mathbf{w}_{t+1})\|_2^2.
\end{aligned}
$$

Since

$$
\begin{aligned}
\|\mathbf{v}_t - \nabla F(\mathbf{w}_{t+1})\|_2^2 &= \|\mathbf{v}_t - \nabla F(\mathbf{w}_t) + \nabla F(\mathbf{w}_t) - \nabla F(\mathbf{w}_{t+1})\|_2^2 \\
&\leq 2\|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|_2^2 + 2\|\nabla F(\mathbf{w}_t) - \nabla F(\mathbf{w}_{t+1})\|_2^2 \\
&\leq 2\|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|_2^2 + 2L_F^2\|\mathbf{w}_t) - \mathbf{w}_{t+1}\|_2^2.
\end{aligned}
$$

Due to $2L_F^2 \leq \frac{L_F}{2\eta_t}$, we have

$$\begin{aligned}
&\|\mathbf{v}_t - \nabla F(\mathbf{w}_{t+1}) + \frac{1}{\eta_t}(\mathbf{w}_{t+1} - \mathbf{w}_t)\|_2^2 \\
&\leq \frac{2(\bar{F}(\mathbf{w}_t) - \bar{F}(\mathbf{w}_{t+1}))}{\eta_t} + \frac{3.5L_F}{\eta_t}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 + 2\|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|_2^2.
\end{aligned}$$

Multiplying both sides by η_t, we have

$$\begin{aligned}
&\eta_t\|\mathbf{v}_t - \nabla F(\mathbf{w}_{t+1}) + \frac{1}{\eta_t}(\mathbf{w}_{t+1} - \mathbf{w}_t)\|_2^2 \\
&\leq 2(\bar{F}(\mathbf{w}_t) - \bar{F}(\mathbf{w}_{t+1})) + 3.5L_F\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 + 2\eta_t\|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|_2^2.
\end{aligned}$$

Adding this inequality to (4.43) gives

$$\begin{aligned}
&\eta_t\|\mathbf{v}_t - \nabla F(\mathbf{w}_{t+1}) + \frac{1}{\eta_t}(\mathbf{w}_{t+1} - \mathbf{w}_t)\|_2^2 + \frac{1}{8\eta_t}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 \\
&\leq 3(\bar{F}(\mathbf{w}_t) - \bar{F}(\mathbf{w}_{t+1})) + 3\eta_t\|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|_2^2 + 3.5L_F\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2.
\end{aligned}$$

Applying (4.43) again to the RHS, we have

$$\begin{aligned}
&\eta_t\|\mathbf{v}_t - \nabla F(\mathbf{w}_{t+1}) + \frac{1}{\eta_t}(\mathbf{w}_{t+1} - \mathbf{w}_t)\|_2^2 + \frac{1}{8\eta_t}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 \\
&\leq (3 + 28L_F\eta_t)(\bar{F}(\mathbf{w}_t) - \bar{F}(\mathbf{w}_{t+1})) + (3\eta_t + 28\eta_t^2 L_F)\|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|_2^2 \\
&\leq 10(\bar{F}(\mathbf{w}_t) - \bar{F}(\mathbf{w}_{t+1})) + 10\eta_t\|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|_2^2.
\end{aligned}$$

Combining this with (4.39), we finish the proof. □

Since the above lemma resembles that in Lemma 4.9, hence, it remains a simple exercise to derive the complexity of using the MA estimator similar to Theorem 4.3 and of using the STORM estimator similar to Theorem 4.4.

Corollary 4.1 *Consider the method (4.37). Under the same assumptions and similar settings as in Theorem 4.3, the method finds an ϵ-stationary solution with a complexity of $O(1/\epsilon^4)$. Under the same assumptions and similar settings as in Theorem 4.4, the method finds an ϵ-stationary solution with a complexity of $O(1/\epsilon^3)$.*

Why it matters

Since standard regularized stochastic optimization $\mathbb{E}_\zeta[g(\mathbf{w};\zeta)] + r(\mathbf{w})$ is a special case, the above results directly apply. This corollary shows that regularized problems can be solved with the same complexities as unregularized ones by employing either the moving-average gradient estimator or the STORM gradient estimator. In contrast, without these estimators, solving non-convex regularized problems requires a large batch size at every iteration (Lan, 2020)[Section 6.2.3].

4.5 Structured Optimization with Compositional Gradient

In this section, we extend the compositional optimization technique to address other structured optimization problems, including min-max optimization, min-min optimization, and bilevel optimization. These problems share a common structure in the form of a compositional gradient, denoted by $\mathcal{M}(\mathbf{w}, \mathbf{u}^*(\mathbf{w}))$, where $\mathcal{M}$ is a mapping that is Lipschitz continuous with respect to its second argument, and $\mathbf{u}^*(\mathbf{w})$ is defined as the solution to a strongly convex optimization problem:

$$\mathbf{u}^*(\mathbf{w}) = \arg\min_{\mathbf{u}\in\mathcal{U}} h(\mathbf{w}, \mathbf{u}). \tag{4.44}$$

This structure generalizes the gradient of a compositional function $f(g(\mathbf{w}))$, whose gradient takes the form $\mathcal{M}(\mathbf{w}, \mathbf{u}^*(\mathbf{w})) = \nabla g(\mathbf{w})\nabla f(\mathbf{u}^*(\mathbf{w}))$ with

$$\mathbf{u}^*(\mathbf{w}) = \arg\min_{\mathbf{u}} \|\mathbf{u} - g(\mathbf{w})\|_2^2.$$

The high-level idea underlying the algorithms and analysis presented below is summarized as follows. To estimate $\mathcal{M}(\mathbf{w}, \mathbf{u}^*(\mathbf{w}))$ at $\mathbf{w}_t$, we use an auxiliary variable $\mathbf{u}_t$ to track the optimal solution $\mathbf{u}^*(\mathbf{w}_t)$, which is defined by solving (4.44) with one step update at $\mathbf{w}_t$. A key aspect of the analysis is that the error in the approximation of $\mathcal{M}(\mathbf{w}_t, \mathbf{u}_t)$ is controlled by the estimation error $\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\|_2$, due to the Lipschitz continuity of $\mathcal{M}$:

$$\|\mathcal{M}(\mathbf{w}_t, \mathbf{u}_t) - \mathcal{M}(\mathbf{w}_t, \mathbf{u}^*(\mathbf{w}_t))\|_2^2 \leq O(\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\|_2^2). \tag{4.45}$$

Moreover, since $\mathbf{u}^*(\mathbf{w})$ is the solution to a strongly convex problem and is Lipschitz continuous with respect to $\mathbf{w}$, we can construct a recursion for $\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\|_2^2$ to effectively bound the cumulative error over iterations.

In cases where $\mathcal{M}(\mathbf{w}_t, \mathbf{u}_t)$ cannot be computed exactly and is instead approximated by a stochastic estimator $\mathcal{M}(\mathbf{w}_t, \mathbf{u}_t; \zeta_t)$, where ζ_t is a random variable, we employ a moving average (MA) estimator:

$$\mathbf{v}_t = (1 - \beta_t)\mathbf{v}_{t-1} + \beta_t \, \mathcal{M}(\mathbf{w}_t, \mathbf{u}_t; \zeta_t).$$

The model update is then performed using:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{v}_t.$$

Alternatively, if $\mathcal{M}(\mathbf{w}_t, \mathbf{u}_t)$ is directly computable, the update simplifies to:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \, \mathcal{M}(\mathbf{w}_t, \mathbf{u}_t).$$

4.5.1 Non-convex Min-Max Optimization

We consider a non-convex min-max optimization problem:

$$\min_{\mathbf{w}\in\mathbb{R}^d} \max_{\mathbf{u}\in\mathcal{U}} f(\mathbf{w},\mathbf{u}) := \mathbb{E}_\xi[f(\mathbf{w},\mathbf{u};\xi)], \tag{4.46}$$

where $f(\mathbf{w},\mathbf{u})$ is a continuous and differentiable and $\mathcal{U}$ is a closed convex set. Let $F(\mathbf{w}) = \max_{\mathbf{u}\in\mathcal{U}} f(\mathbf{w},\mathbf{u})$. Denote by $\nabla_1 f(\cdot,\cdot)$ and $\nabla_2 f(\cdot,\cdot)$ the partial gradients of the first and second variable, respectively.

We make the following assumptions.

Assumption 4.8. *Regarding the problem (4.46), the following conditions hold:*

(i) $f(\mathbf{w},\mathbf{u})$ *is* μ*-strongly concave in terms of* $\mathbf{u}$*, and* $\mathbf{u}^*(\mathbf{w}) = \arg\max_{\mathbf{u}\in\mathcal{U}} f(\mathbf{w},\mathbf{u})$ *exists for any* $\mathbf{w}$.

(ii) $\nabla_1 f(\mathbf{w},\mathbf{u})$ *is* L_1*-Lipschitz continuous such that*

$$\|\nabla_1 f(\mathbf{w},\mathbf{u}) - \nabla_1 f(\mathbf{w}',\mathbf{u}')\|_2 \leq L_1(\|\mathbf{w}-\mathbf{w}'\|_2 + \|\mathbf{u}-\mathbf{u}'\|_2). \tag{4.47}$$

(iii) $\nabla_2 f(\mathbf{w},\mathbf{u})$ *is* L_{21}*-Lipschitz continuous with respect to the first variable and is* L_{22}*-Lipschitz continuous with respect to the second variable*

$$\|\nabla_2 f(\mathbf{w},\mathbf{u}) - \nabla_2 f(\mathbf{w}',\mathbf{u}')\|_2 \leq L_{21}\|\mathbf{w}-\mathbf{w}'\|_2 + L_{22}\|\mathbf{u}-\mathbf{u}'\|_2. \tag{4.48}$$

(iv) there exist σ_1, σ_2 *such that*

$$\mathbb{E}[\|\nabla_1 f(\mathbf{w},\mathbf{u};\xi) - \nabla_1 f(\mathbf{w},\mathbf{u})\|_2^2] \leq \sigma_1^2, \tag{4.49}$$

$$\mathbb{E}[\|\nabla_2 f(\mathbf{w},\mathbf{u};\xi) - \nabla_2 f(\mathbf{w},\mathbf{u})\|_2^2] \leq \sigma_2^2. \tag{4.50}$$

(v) $F_* = \min_{\mathbf{w}} F(\mathbf{w}) \geq -\infty$.

4.5.1.1 A Double-loop Large mini-batch method

Let us first consider a straightforward approach that updates $\mathbf{w}_t$ using a large-batch gradient estimator

$$\mathbf{v}_t = \frac{1}{B}\sum_{i=1}^{B} \nabla_1 f(\mathbf{w}_t,\mathbf{u}_t;\zeta_{i,t}),$$

and computes $\mathbf{u}_t$ via an inner-loop SGD with K updates. It suffices to have $K = O(L_1^2\sigma_2^2/(\mu^2\epsilon^2))$ (by Lemma 3.8) such that

$$\mathbb{E}[\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\|_2^2] \leq \frac{\epsilon^2}{L_1^2}.$$

If $B = O(\sigma_1^2/\epsilon^2)$, following the Lemma 4.18 below we have

Algorithm 12 SMDA

1: **Input:** learning rate schedules $\{\eta_t\}_{t=1}^T$, $\{\gamma_t\}_{t=1}^T$, $\{\beta_t\}_{t=1}^T$; starting points $\mathbf{w}_0, \mathbf{u}_1, \mathbf{v}_0$
2: $\mathbf{w}_1 = \mathbf{w}_0 - \eta_0 \mathbf{v}_0$
3: **for** $t = 1, \ldots, T$ **do**
4: Sample ζ_t
5: Update $\mathbf{u}_{t+1} = \Pi_{\mathcal{U}}[\mathbf{u}_t + \gamma_t \nabla_2 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t)]$
6: Compute the vanilla gradient estimator $\mathbf{z}_t = \nabla_1 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t)$
7: Update the MA gradient estimator $\mathbf{v}_t = (1 - \beta_t)\mathbf{v}_{t-1} + \beta_t \mathbf{z}_t$
8: Update the model by $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{v}_t$
9: **end for**

$$\mathbb{E}[\|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|_2^2] \leq \mathbb{E}\left[\left\|\frac{1}{B}\sum_{i=1}^{B} \nabla_1 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_{i,t}) - \nabla_1 f(\mathbf{w}_t, \mathbf{u}^*(\mathbf{w}_t))\right\|_2^2\right]$$
$$\leq O\left(\frac{\sigma_1^2}{B} + L_1^2 \|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\|_2^2\right) \leq \epsilon^2.$$

Combining this with Lemma 4.9, we can set the step size $\eta_t = O(1/L_F)$ and the number of iterations $T = O(L_F/\epsilon^2)$, yielding an overall sample complexity of

$$BT + KT = O\left(\frac{L_F \sigma_1^2}{\epsilon^4} + \frac{L_F L_1^2 \sigma_2^2}{\mu^2 \epsilon^4}\right).$$

4.5.1.2 A Stochastic Momentum Method

We present a solution method in Algorithm 12, referred to as **SMDA** (Stochastic Momentum Descent-Ascent). The method begins by updating the dual variable using stochastic gradient ascent (Step 4), then computes the moving average gradient estimator $\mathbf{v}_t$ for the primal variable (Step 6), and finally updates the primal variable using this estimator (Step 7). When $\beta_t = 1$, the method reduces to SGDA. However, setting $\beta_t < 1$ is crucial for achieving improved complexity. Conceptually, the method shares similarities with SCMA.

Convergence Analysis

We will prove the convergence of the gradient norm of $F(\mathbf{w})$. We first prove the following lemmas.

Lemma 4.17 *Let* $\mathbf{u}^*(\mathbf{w}) = \arg\max_{\mathbf{u}\in\mathcal{U}} f(\mathbf{w}, \mathbf{u})$. *Under Assumption 4.8(i), (iii),* $\mathbf{u}^*(\cdot)$ *is* κ*-Lipschitz continuous with* $\kappa = \frac{L_{21}}{\mu}$.

Proof. Let us consider $\mathbf{w}_1, \mathbf{w}_2$. By the optimality condition of $\mathbf{u}^*(\mathbf{w}_1)$ and $\mathbf{u}^*(\mathbf{w}_2)$ for a concave function, we have

$$\nabla_2 f(\mathbf{w}_1, \mathbf{u}^*(\mathbf{w}_1))^\top (\mathbf{u} - \mathbf{u}^*(\mathbf{w}_1)) \le 0, \quad \forall \mathbf{u} \in \mathcal{U}$$
$$\nabla_2 f(\mathbf{w}_2, \mathbf{u}^*(\mathbf{w}_2))^\top (\mathbf{u} - \mathbf{u}^*(\mathbf{w}_2)) \le 0, \quad \forall \mathbf{u} \in \mathcal{U}.$$

Let $\mathbf{u} = \mathbf{u}^*(\mathbf{w}_2)$ in the first inequality and $\mathbf{u} = \mathbf{u}^*(\mathbf{w}_1)$ in the second equality and add them together we have

$$(\nabla_2 f(\mathbf{w}_1, \mathbf{u}^*(\mathbf{w}_1)) - \nabla_2 f(\mathbf{w}_2, \mathbf{u}^*(\mathbf{w}_2)))^\top (\mathbf{u}^*(\mathbf{w}_2) - \mathbf{u}^*(\mathbf{w}_1)) \le 0.$$

Since $-f(\mathbf{w}_1, \cdot)$ is μ-strongly convex, due to Lemma 1.6, we have

$$(\nabla_2 f(\mathbf{w}_1, \mathbf{u}^*(\mathbf{w}_1)) - \nabla_2 f(\mathbf{w}_1, \mathbf{u}^*(\mathbf{w}_2)))^\top (\mathbf{u}^*(\mathbf{w}_2) - \mathbf{u}^*(\mathbf{w}_1)) \ge \mu \|\mathbf{u}^*(\mathbf{w}_2) - \mathbf{u}^*(\mathbf{w}_1)\|_2^2.$$

Combining these two inequalities we have

$$\begin{aligned}\mu \|\mathbf{u}^*(\mathbf{w}_2) - \mathbf{u}^*(\mathbf{w}_1)\|_2^2 &\le (\nabla_2 f(\mathbf{w}_2, \mathbf{u}^*(\mathbf{w}_2)) - \nabla_2 f(\mathbf{w}_1, \mathbf{u}^*(\mathbf{w}_2)))^\top (\mathbf{u}^*(\mathbf{w}_2) - \mathbf{u}^*(\mathbf{w}_1)) \\ &\le \|\nabla_2 f(\mathbf{w}_2, \mathbf{u}^*(\mathbf{w}_2)) - \nabla_2 f(\mathbf{w}_1, \mathbf{u}^*(\mathbf{w}_2))\|_2 \|\mathbf{u}^*(\mathbf{w}_2) - \mathbf{u}^*(\mathbf{w}_1)\|_2 \\ &\le L_{21} \|\mathbf{w}_2 - \mathbf{w}_1\|_2 \|\mathbf{u}^*(\mathbf{w}_2) - \mathbf{u}^*(\mathbf{w}_1)\|_2.\end{aligned}$$

Thus,

$$\|\mathbf{u}^*(\mathbf{w}_2) - \mathbf{u}^*(\mathbf{w}_1)\|_2 \le \frac{L_{21}}{\mu} \|\mathbf{w}_2 - \mathbf{w}_1\|_2.$$

□

Lemma 4.18 *Under Assumption 4.8(i) and (ii), $\nabla F(\mathbf{w}) = \nabla_1 f(\mathbf{w}, \mathbf{u}^*(\mathbf{w}))$, and it is is L_F-Lipschitz continuous with $L_F = L_1(1 + \kappa)$.*

Proof. If $\mathcal{U}$ is bounded, the Danskin's theorem implies that $\nabla F(\mathbf{w}) = \nabla_1 f(\mathbf{w}, \mathbf{u}^*(\mathbf{w}))$. If $\mathcal{U}$ is unbounded, we have

$$\nabla F(\mathbf{w}) = \nabla_1 f(\mathbf{w}, \mathbf{u}^*(\mathbf{w})) + \frac{\partial u^*(\mathbf{w})}{\partial \mathbf{w}}^\top \nabla_2 f(\mathbf{w}, \mathbf{u}^*(\mathbf{w})) = \nabla_1 f(\mathbf{w}, \mathbf{u}^*(\mathbf{w})), \quad (4.51)$$

where the last equality follows from $\nabla_2 f(\mathbf{w}, \mathbf{u}^*(\mathbf{w})) = 0$. To establish the Lipschitz continuity of $\nabla F(\mathbf{w})$, let us consider $\mathbf{w}_1$ and $\mathbf{w}_2$. We have

$$\begin{aligned}&\|\nabla F(\mathbf{w}_1) - \nabla F(\mathbf{w}_2)\|_2 = \|\nabla_1 f(\mathbf{w}_1, \mathbf{u}^*(\mathbf{w}_1)) - \nabla_1 f(\mathbf{w}_2, \mathbf{u}^*(\mathbf{w}_2))\|_2 \\ &\le L_1(\|\mathbf{w}_1 - \mathbf{w}_2\|_2 + \|\mathbf{u}^*(\mathbf{w}_1) - \mathbf{u}^*(\mathbf{w}_2)\|_2) \le L_1(1 + \kappa)\|\mathbf{w}_1 - \mathbf{w}_2\|_2.\end{aligned}$$

□

Next, we prove two lemmas similar to Lemma 4.8 and Lemma 4.1, regarding the recursion of gradient estimation error and the estimation error of $\mathbf{u}$, respectively. The descent lemma (Lemma 4.9) still holds.

Lemma 4.19 *It holds that*

$$\mathbb{E}_{\xi_t}\left[\|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|_2^2\right] \le (1-\beta_t)\|\mathbf{v}_{t-1} - \nabla F(\mathbf{w}_{t-1})\|_2^2 + \frac{2L_F^2}{\beta_t}\|\mathbf{w}_{t-1} - \mathbf{w}_t\|_2^2 + 4L_1^2\beta_t\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\|_2^2 + \beta_t^2\sigma_1^2.$$

Proof. Let $\mathbf{z}_t = \nabla_1 f(\mathbf{w}_t, \mathbf{u}_t; \xi_t)$ and $\mathcal{M}_t = \mathbb{E}_t[\mathbf{z}_t] = \nabla_1 f(\mathbf{w}_t, \mathbf{u}_t)$. Then $\mathbf{v}_t = (1-\beta_t)\mathbf{v}_{t-1} + \beta_t \mathbf{z}_t$. Noting that $\mathbb{E}_t[\|\mathbf{z}_t - \mathcal{M}_t\|_2^2] \le \sigma_1^2$ and $\|\mathcal{M}_t - \nabla F(\mathbf{w}_t)\|_2^2 \le L_1^2\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w})\|_2^2$. Plugging these into Lemma 4.7 finishes the proof. □

Lemma 4.20 *Suppose Assumption 4.8 (i), (iii), (iv) hold. Consider the update* $\mathbf{u}_t = \Pi_{\mathcal{U}}[\mathbf{u}_t + \gamma_t \nabla_2 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t)]$. *If* $\gamma_t < 1/L_{22} < 1/\mu$, *we have*

$$\mathbb{E}_t[\|\mathbf{u}_{t+1} - \mathbf{u}^*(\mathbf{w}_{t+1})\|_2^2] \le (1 - \frac{\gamma_t\mu}{2})\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\|_2^2 + \frac{3\kappa^2}{\gamma_t\mu}\mathbb{E}_t[\|\mathbf{w}_t - \mathbf{w}_{t+1}\|_2^2] + 2\gamma_t^2\sigma_2^2.$$

Proof. By Lemma 3.7, if $\gamma < 1/L_{22}$ we have

$$\mathbb{E}_t[\|\mathbf{u}_{t+1} - \mathbf{u}^*(\mathbf{w}_t)\|_2^2] \le (1-\gamma_t\mu)\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\|_2^2 + \gamma_t^2\sigma_2^2. \tag{4.52}$$

Then,

$$\begin{aligned}
&\mathbb{E}_t[\|\mathbf{u}_{t+1} - \mathbf{u}^*(\mathbf{w}_{t+1})\|_2^2] \le (1 + \frac{\gamma_t\mu}{2})\mathbb{E}_t[\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\|_2^2] \\
&+ (1 + \frac{2}{\gamma_t\mu})\mathbb{E}_t[\|\mathbf{u}^*(\mathbf{w}_t) - \mathbf{u}^*(\mathbf{w}_{t+1})\|_2^2] \\
&\le (1 + \frac{\gamma_t\mu}{2})(1-\gamma_t\mu)\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\|_2^2 + (1 + \frac{\gamma_t\mu}{2})\gamma_t^2\sigma_2^2 \\
&+ \frac{2+\gamma_t\mu}{\gamma_t\mu}\kappa^2\mathbb{E}_t[\|\mathbf{w}_t - \mathbf{w}_{t+1}\|_2^2] \\
&\le (1 - \frac{\gamma_t\mu}{2})\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\|_2^2 + 2\gamma_t^2\sigma_2^2 + \frac{3\kappa^2}{\gamma_t\mu}\mathbb{E}_t[\|\mathbf{w}_t - \mathbf{w}_{t+1}\|_2^2],
\end{aligned}$$

where the first inequality uses the Young's inequality, and the last inequality uses $\gamma\mu < 1$. □

Finally, we can prove the following theorem regarding the convergence of SMDA.

Theorem 4.5 *Suppose Assumption 4.8 holds. By setting* $\beta_t = \beta = \epsilon^2/(3\sigma_1^2)$, $\gamma_t = \gamma = \mu\epsilon^2/(96L_1^2\sigma_2^2)$ *and* $\eta_t = \eta = \min(\frac{\beta}{\sqrt{8}L_F}, \frac{\gamma\mu}{16\sqrt{3}L_1\kappa}, \frac{1}{2L_F})$ *in SMDA, then the following holds*

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1}\left\{\frac{1}{4}\|\mathbf{v}_t\|_2^2 + \|\nabla F(\mathbf{w}_t)\|_2^2\right\}\right] \le \epsilon^2, \tag{4.53}$$

with an iteration complexity of

$$T = O\left(\max\left\{\frac{C_\Upsilon L_F}{\epsilon^2}, \frac{C_\Upsilon \sigma_1^2 L_F}{\epsilon^4}, \frac{C_\Upsilon L_1^3 \kappa \sigma_2^2}{\epsilon^4 \mu^2}\right\}\right), \tag{4.54}$$

where $C_\Upsilon = 2(F(\mathbf{w}_0) - F_*) + \frac{1}{\sqrt{8}L_F}\|\mathbf{v}_0 - \nabla F(\mathbf{w}_0)\|_2^2 + \frac{L_1}{\sqrt{3}\kappa}\|\mathbf{u}_0 - \mathbf{u}^*(\mathbf{w}_0)\|_2^2$.

Why it matters

The MA gradient estimator in SMDA is critical to obtaining a complexity of $O(1/\epsilon^4)$. If we simply update the primal variable by SGD, the algorithm becomes SGDA. The convergence analysis of SGDA for non-convex minimax problems will suffer from a large batch size issue or slow convergence. In particular, SGDA with a batch size of $O(1/\epsilon^2)$ can find an ϵ-stationary solution in $O(1/\epsilon^2)$ iterations when the problem is smooth in terms of primal and dual variables and strongly-concave in terms of dual variable, yielding a sample complexity of $O(1/\epsilon^4)$. If using a constant batch size $O(1)$, SGDA may need $O(1/\epsilon^8)$ iterations for finding an ϵ-stationary solution (Lin et al., 2020).

Proof. The proof is similar to Theorem 4.3. Let us see the three inequalities in Lemma 4.9, Lemma 4.19, and 4.20 that we have proved so far:

$$(*) F(\mathbf{w}_{t+1}) \leq F(\mathbf{w}_t) + \frac{\eta}{2}\|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\|_2^2 - \frac{\eta}{2}\|\nabla F(\mathbf{w}_t)\|_2^2 - \frac{\eta}{4}\|\mathbf{v}_t\|_2^2,$$

$$(\sharp)\mathbb{E}\left[\|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|_2^2\right] \leq \mathbb{E}\left[(1-\beta)\|\mathbf{v}_{t-1} - \nabla F(\mathbf{w}_{t-1})\|_2^2 + \frac{2L_F^2\eta^2}{\beta}\|\mathbf{v}_{t-1}\|_2^2\right]$$
$$+ 4L_1^2\beta\mathbb{E}\left[\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\|_2^2 + \beta^2\sigma_1^2\right],$$

$$(\diamond)\mathbb{E}\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\|_2^2 \leq \mathbb{E}\left[(1 - \frac{\gamma\mu}{2})\|\mathbf{u}_{t-1} - \mathbf{u}^*(\mathbf{w}_{t-1})\|_2^2 + 2\gamma^2\sigma_2^2 + \frac{3\kappa^2\eta^2}{\gamma\mu}\|\mathbf{v}_{t-1}\|_2^2\right].$$

Let $\bar{\gamma} = \gamma\mu/2$, the last inequality becomes:

$$(\diamond)\mathbb{E}\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\|_2^2 \leq \mathbb{E}\left[(1-\bar{\gamma})\|\mathbf{u}_{t-1} - \mathbf{u}^*(\mathbf{w}_{t-1})\|_2^2 + 8\bar{\gamma}^2\frac{\sigma_2^2}{\mu^2} + \frac{3\kappa^2\eta^2}{2\bar{\gamma}}\|\mathbf{v}_{t-1}\|_2^2\right].$$

Let us define $A_t = 2(F(\mathbf{w}_t) - F_*)$ and $B_t = \|\nabla F(\mathbf{w}_t)\|_2^2$, $\Gamma_t = \|\mathbf{v}_t\|_2^2/2$, $\Delta_t = \|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\|_2^2$, $\delta_t = \|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\|_2^2$. Then the three inequalities $(*)$, $(\sharp)$, $(\diamond)$ satisfy that in Lemma 4.10 with $C_1 = 4L_1^2, C_2 = 2L_F^2, C_3 = 3\kappa^2/2, \sigma^2 = \sigma_1^2, \sigma'^2 = 8\sigma_2^2/\mu^2$. If $\eta, \beta, \bar{\gamma}$ satisfy

$$\beta = \frac{\epsilon^2}{3\sigma^2} = \frac{\epsilon^2}{3\sigma_1^2}, \quad \bar{\gamma} = \frac{\epsilon^2}{6C_1\sigma'^2} = \frac{\epsilon^2\mu^2}{192L_1^2\sigma_2^2},$$
$$\eta = \min(\frac{1}{2L_F}, \frac{\beta}{\sqrt{4C_2}}, \frac{\bar{\gamma}}{\sqrt{8C_1C_3}}) = \min(\frac{1}{2L_F}, \frac{\beta}{\sqrt{8}L_F}, \frac{\bar{\gamma}}{\sqrt{48}L_1\kappa}),$$

then (4.89) holds, and the iteration complexity becomes

$$T = O\left(\max\left\{\frac{C_\Upsilon L_F}{\epsilon^2}, \frac{C_\Upsilon\sigma^2\sqrt{C_2}}{\epsilon^4}, \frac{C_\Upsilon\sqrt{C_1C_3}C_1\sigma'^2}{\epsilon^4}\right\}\right)$$
$$= O\left(\max\left\{\frac{C_\Upsilon L_F}{\epsilon^2}, \frac{C_\Upsilon\sigma_1^2 L_F}{\epsilon^4}, \frac{C_\Upsilon L_1^3\kappa\sigma_2^2}{\epsilon^4\mu^2}\right\}\right).$$

□

Critical: It is worth mentioning that an improved complexity of $O(1/\epsilon^3)$ can be achieved by employing the STORM gradient estimator for both the primal and dual variables under the mean-square smooth condition of the objective.

4.5.2 Non-convex Min-Min Optimization

We can extend SMDA to solving a non-convex strongly-convex min-min problem:

$$\min_{\mathbf{w}\in\mathbb{R}^d}\min_{\mathbf{u}\in\mathcal{U}} f(\mathbf{w},\mathbf{u}) := \mathbb{E}_\xi[f(\mathbf{w},\mathbf{u};\xi)], \tag{4.55}$$

where $f(\mathbf{w},\mathbf{u})$ is smooth, non-convex in terms of $\mathbf{w}$ and strongly convex in terms of $\mathbf{u}$ and $\mathcal{U}$ is a closed convex set. If the $\mathbf{u}^*(\mathbf{w}) = \arg\min_{\mathbf{u}\in\mathcal{U}} f(\mathbf{w},\mathbf{u})$ exists and unique, then we have $\nabla F(\mathbf{w}) = \nabla_1 f(\mathbf{w},\mathbf{u}^*(\mathbf{w}))$. Hence, its gradient also exhibits a compositional structure, where the inner function $\mathbf{u}^*(\mathbf{w})$ is a solution to a strongly convex problem.

SMDA can be modified by replacing the $\mathbf{u}$ update with

$$\mathbf{u}_{t+1} = \Pi_{\mathcal{U}}[\mathbf{u}_t - \gamma_t\nabla_2 f(\mathbf{w}_t,\mathbf{u}_t;\zeta_t)].$$

Then, the same convergence result in the last subsection can be established for min-min problem, which is omitted here.

4.5.2.1 Application to weakly convex minimization

Next, we present an application to solving weakly convex minimization problems:

Algorithm 13 A novel method for weakly convex minimization

Input: learning rate schedules $\{\eta_t\}_{t=1}^T$, $\{\gamma_t\}_{t=1}^T$; starting points $\mathbf{w}_1, \mathbf{u}_1, \mathbf{v}_1$
for $t = 1, \ldots, T$ **do**
    Sample ζ_t and compute $\mathcal{G}(\mathbf{u}_t; \zeta_t) = \partial g(\mathbf{u}_t; \zeta_t)$
    Update $\mathbf{u}_{t+1} = \mathbf{u}_t - \gamma_t(\mathcal{G}(\mathbf{u}_t; \zeta_t) + \rho(\mathbf{u}_t - \mathbf{w}_t))$
    Update $\mathbf{w}_{t+1} = (1 - 2\eta_t\rho)\mathbf{w}_t + 2\eta_t\rho\mathbf{u}_t$
end for

$$\min_{\mathbf{w}} F(\mathbf{w}) := \mathbb{E}[g(\mathbf{w}; \zeta)], \tag{4.56}$$

where $F > -\infty$ is ρ-weakly convex, as discussed in Chapter 3.

As argued in Section 3.1.4, an ϵ-stationary solution of the Moreau envelope of $F(\mathbf{w})$ corresponds to a nearly ϵ-stationary solution of the original problem. Hence, we consider optimizing the Moreau envelope directly:

$$\min_{\mathbf{w}} F_\rho(\mathbf{w}) := \min_{\mathbf{u}} \mathbb{E}[g(\mathbf{u}; \zeta)] + \rho\|\mathbf{u} - \mathbf{w}\|_2^2. \tag{4.57}$$

Define $f(\mathbf{w}, \mathbf{u}) = \mathbb{E}[g(\mathbf{u}; \zeta)] + \rho\|\mathbf{u} - \mathbf{w}\|_2^2$. Then $f(\mathbf{w}, \mathbf{u})$ is ρ-strongly convex with respect to $\mathbf{u}$ due to the ρ-weak convexity of F.

For updating $\mathbf{u}$, we use the standard SGD:

$$\mathbf{u}_{t+1} = \mathbf{u}_t - \gamma_t(\mathcal{G}(\mathbf{u}_t; \zeta_t) + 2\rho(\mathbf{u}_t - \mathbf{w}_t)). \tag{4.58}$$

where $\mathcal{G}(\mathbf{u}_t; \zeta_t) \in \partial g(\mathbf{u}_t; \zeta_t)$. For updating $\mathbf{w}$, then we just apply GD with its gradient given by $\nabla_1 f(\mathbf{w}_t, \mathbf{u}_t) = 2\rho(\mathbf{w}_t - \mathbf{u}_t)$:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t 2\rho(\mathbf{w}_t - \mathbf{u}_t) = (1 - 2\eta_t\rho)\mathbf{w}_t + 2\eta_t\rho\mathbf{u}_t. \tag{4.59}$$

We present the updates in Algorithm 13. An interesting observation about this algorithm is that the $\mathbf{u}$ update is similar to the Momentum update (4.18) except that the momentum term $\mathbf{u}_t - \mathbf{u}_{t-1}$ is replaced by $\mathbf{u}_t - \mathbf{w}_t$, where $\mathbf{w}_t$ is a MA weight vector.

Convergence Analysis

Let us first prove the following lemma.

Lemma 4.21 *We have (i) F_ρ is L_F-smooth with $L_F = \frac{6}{\rho}$; (ii) $\nabla_1 f(\mathbf{w}, \mathbf{u})$ is Lipschitz continuous with $L_1 = 2\rho$, and (iii) $\mathbf{u}^*(\mathbf{w})$ is 1-Lipschitz continuous.*

Proof. The smoothness of F_ρ has been proved in Proposition 3.1 with $\lambda = \rho/2$. The Lipschitz continuity of $\nabla_1 f(\mathbf{w}, \mathbf{u}) = 2\rho(\mathbf{w} - \mathbf{u})$ is obvious. Next, let us prove the Lipschitz continuity of $\mathbf{u}^*(\mathbf{w})$. The proof is similar to that of Lemma 4.17.

Let us consider $\mathbf{w}_1, \mathbf{w}_2$. By the optimality condition of $\mathbf{u}^*(\mathbf{w}_1)$ and $\mathbf{u}^*(\mathbf{w}_2)$ for a concave function, there exists $\mathbf{v}(\mathbf{w}_1) \in \partial_2 f(\mathbf{w}_1, \mathbf{u}^*(\mathbf{w}_1)), \mathbf{v}(\mathbf{w}_2) \in \partial_2 f(\mathbf{w}_2, \mathbf{u}^*(\mathbf{w}_2))$

$$\mathbf{v}(\mathbf{w}_1)^\top(\mathbf{u} - \mathbf{u}^*(\mathbf{w}_1)) \le 0, \quad \forall \mathbf{u}$$
$$\mathbf{v}(\mathbf{w}_2)^\top(\mathbf{u} - \mathbf{u}^*(\mathbf{w}_2)) \le 0, \quad \forall \mathbf{u}$$

Let $\mathbf{u} = \mathbf{u}^*(\mathbf{w}_2)$ in the first inequality and $\mathbf{u} = \mathbf{u}^*(\mathbf{w}_1)$ in the second equality and add them together we have

$$(\mathbf{v}(\mathbf{w}_1) - \mathbf{v}(\mathbf{w}_2))^\top(\mathbf{u}^*(\mathbf{w}_2) - \mathbf{u}^*(\mathbf{w}_1)) \le 0.$$

Since $-f(\mathbf{w}_1, \cdot)$ is ρ-strongly convex, similar to Lemma 1.6, we have for any $\mathbf{v} \in \partial_2 f(\mathbf{w}_1, \mathbf{u}^*(\mathbf{w}_2))$,

$$(\mathbf{v}(\mathbf{w}_1) - \mathbf{v})^\top(\mathbf{u}^*(\mathbf{w}_2) - \mathbf{u}^*(\mathbf{w}_1)) \ge \rho\|\mathbf{u}^*(\mathbf{w}_2) - \mathbf{u}^*(\mathbf{w}_1)\|_2^2.$$

Combining these two inequalities we have

$$\begin{aligned}\rho\|\mathbf{u}^*(\mathbf{w}_2) - \mathbf{u}^*(\mathbf{w}_1)\|_2^2 &\le (\mathbf{v}(\mathbf{w}_2) - \mathbf{v})^\top(\mathbf{u}^*(\mathbf{w}_2) - \mathbf{u}^*(\mathbf{w}_1)) \\ &\le \|\mathbf{v}(\mathbf{w}_2) - \mathbf{v}\|_2\|\mathbf{u}^*(\mathbf{w}_2) - \mathbf{u}^*(\mathbf{w}_1)\|_2.\end{aligned}$$

Since there exists $\mathbf{v}' \in \partial g(\mathbf{u}^*(\mathbf{w}_2))$ such that $\mathbf{v}(\mathbf{w}_2) = \mathbf{v}' + \rho(\mathbf{u}^*(\mathbf{w}_2) - \mathbf{w}_2)$, we let $\mathbf{v} = \mathbf{v}' + \rho(\mathbf{u}^*(\mathbf{w}_2) - \mathbf{w}_1)$, then

$$\|\mathbf{u}^*(\mathbf{w}_2) - \mathbf{u}^*(\mathbf{w}_1)\|_2 \le \|\mathbf{w}_2 - \mathbf{w}_1\|_2.$$

□

Since $\partial_2 f(\mathbf{w}, \mathbf{u})$ is not Lipschitz continuous with respect to $\mathbf{u}$, lemma 4.20 is not directly applicable. We develop a similar one below.

Lemma 4.22 *Consider the following update:*

$$\mathbf{u}_{t+1} = \mathbf{u}_t - \gamma_t(\mathcal{G}(\mathbf{u}_t; \zeta_t) + 2\rho(\mathbf{u}_t - \mathbf{w}_t)).$$

If $\mathbb{E}_\zeta[\|\mathcal{G}(\mathbf{u}; \zeta)\|_2^2] \le G^2$ *and* $\gamma_t\rho < 1/8$*, then we have*

$$\begin{aligned}&\mathbb{E}_t\|\mathbf{u}_{t+1} - \mathbf{u}^*(\mathbf{w}_{t+1})\|_2^2 \\ &\le \left(1 - \frac{\gamma_t\rho}{2}\right)\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\|_2^2 + 8\gamma_t^2G^2 + \frac{12}{\gamma_t\rho}\mathbb{E}_t\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2.\end{aligned}$$

Proof. Since $\mathbf{u}_{t+1}$ is one-step SGD update of $f(\mathbf{w}_t, \mathbf{u})$, the proof is similar to Lemma 3.7 for the non-smooth case.

$$\begin{aligned}&\|\mathbf{u}_{t+1} - \mathbf{u}^*(\mathbf{w}_t)\|_2^2 = \|\mathbf{u}_t - \gamma_t\left(\mathcal{G}(\mathbf{u}_t; \zeta_t) + 2\rho(\mathbf{u}_t - \mathbf{w}_t)\right) - \mathbf{u}^*(\mathbf{w}_t)\|_2^2 \qquad (4.60)\\ &= \|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\|_2^2 + \gamma_t^2\|\mathcal{G}(\mathbf{u}_t; \zeta_t) + 2\rho(\mathbf{u}_t - \mathbf{w}_t)\|_2^2 \\ &- 2\gamma_t(\mathcal{G}(\mathbf{u}_t; \zeta_t) + 2\rho(\mathbf{u}_t - \mathbf{w}_t))^\top(\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)).\end{aligned}$$

Note that $0 \in \partial g(\mathbf{u}^*(\mathbf{w}_t)) + 2\rho(\mathbf{u}^*(\mathbf{w}_t) - \mathbf{w}_t)$. Thus, $\mathbf{v}_{t-1} = 2\rho(\mathbf{w}_t - \mathbf{u}^*(\mathbf{w}_t)) \in \partial g(\mathbf{u}^*(\mathbf{w}_t))$,

$$\begin{aligned}\mathbb{E}_t \left\|\mathcal{G}(\mathbf{u}_t;\zeta_t) + 2\rho(\mathbf{u}_t - \mathbf{w}_t)\right\|_2^2 &= \mathbb{E}_t \left\|\mathcal{G}(\mathbf{u}_t;\zeta_t) - \mathbf{v}_{t-1} + 2\rho(\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t))\right\|_2^2 \\ &\le 2(\mathbb{E}_t \left\|\mathcal{G}(\mathbf{u}_t;\zeta_t) + \mathbf{v}_{t-1}\right\|_2 + 8\rho^2 \left\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\right\|_2^2 \\ &\le 8G^2 + 8\rho^2 \left\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\right\|_2^2,\end{aligned}$$

where the last inequality uses $\|\mathbf{v}_{t-1}\|_2 \le G$. For the last term in (4.60), let $\mathbf{v}_{t-1} = \mathbb{E}[\mathcal{G}(\mathbf{u}_t;\zeta_t)] + 2\rho(\mathbf{u}_t - \mathbf{w}_t) \in \partial_2 f(\mathbf{w}_t, \mathbf{u}_t)$, then we have

$$\begin{aligned}&\mathbb{E}_t(\mathcal{G}(\mathbf{u}_t;\zeta_t) + 2\rho(\mathbf{u}_t - \mathbf{w}_t))^\top(\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)) = \mathbf{v}_{t-1}^\top(\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)) \\ &= (\mathbf{v}_{t-1} - \mathbf{v}(\mathbf{w}_t))^\top(\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)) \ge \rho \left\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\right\|_2^2.\end{aligned}$$

where $0 = \mathbf{v}(\mathbf{w}_t) \in \partial_2 f(\mathbf{w}_t, \mathbf{u}^*(\mathbf{w}_t))$ and the last inequality is due to the strong convexity of f in terms of $\mathbf{u}$. Combining the above inequalities we have

$$\begin{aligned}&\left\|\mathbf{u}_{t+1} - \mathbf{u}^*(\mathbf{w}_t)\right\|_2^2 = \left\|\mathbf{u}_t - \gamma_t\left(\partial g(\mathbf{u}_t;\zeta_t) + 2\rho(\mathbf{u}_t - \mathbf{w}_t)\right) - \mathbf{u}^*(\mathbf{w}_t)\right\|_2^2 \\ &\le \left\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\right\|_2^2 + \gamma_t^2(8G^2 + 8\rho^2\left\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\right\|_2^2) - 2\gamma_t\rho\left\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\right\|_2^2 \\ &= (1 - 2\gamma_t\rho + 8\gamma_t^2\rho^2)\left\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\right\|_2^2 + 4\gamma_t^2G^2 \\ &\le (1 - \gamma_t\rho)\left\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\right\|_2^2 + 8\gamma_t^2G^2\end{aligned}$$

where the last inequality uses $\gamma_t \le \frac{1}{8\rho}$. Since $\mathbf{u}^*(\mathbf{w})$ is 1-Lipschitz continuous, we have

$$\begin{aligned}&\mathbb{E}_t \left\|\mathbf{u}_{t+1} - \mathbf{u}^*(\mathbf{w}_{t+1})\right\|_2^2 \\ &\le \left(1 + \frac{\gamma_t\rho}{2}\right)\mathbb{E}_t \left\|\mathbf{u}_{t+1} - \mathbf{u}^*(\mathbf{w}_t)\right\|_2^2 + \left(1 + \frac{2}{\gamma_t\rho}\right)\left\|\mathbf{u}^*(\mathbf{w}_{t+1}) - \mathbf{u}^*(\mathbf{w}_t)\right\|_2^2 \\ &\le \left(1 - \frac{\gamma_t\rho}{2}\right)\left\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\right\|_2^2 + 8\gamma_t^2G^2 + \frac{3}{\gamma_t\rho}\mathbb{E}_t \left\|\mathbf{w}_{t+1} - \mathbf{w}_t\right\|_2^2.\end{aligned}$$

□

Lemma 4.23 *Let $\mathbf{z}_t = 2\rho(\mathbf{w}_t - \mathbf{u}_t)$. For the update $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t\mathbf{z}_t$, if $\eta_t \le 1/(2L_F)$, we have*

$$F_\rho(\mathbf{w}_{t+1}) \le F_\rho(\mathbf{w}_t) + \frac{\eta_t}{2}\left\|\nabla F_\rho(\mathbf{w}_t) - \mathbf{z}_t\right\|_2^2 - \frac{\eta_t}{2}\left\|\nabla F_\rho(\mathbf{w}_t)\right\|_2^2 - \frac{1}{4\eta_t}\left\|\mathbf{w}_{t+1} - \mathbf{w}_t\right\|_2^2,$$

where L_F is the smoothness parameter of $F_\rho(\cdot)$.

Since $\nabla F_\rho(\mathbf{w}_t) = 2\rho(\mathbf{w}_t - \mathbf{u}^*(\mathbf{w}_t))$, hence $\|F_\rho(\mathbf{w}_t) - \mathbf{z}_t\|_2^2 = 4\rho^2\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\|_2^2$, whose recursion has been established in Lemma 4.22. We can combine these two lemmas and establish a complexity of $O(1/\epsilon^4)$ for Algorithm 13 in order to find an ϵ-stationary solution to $F_\rho(\cdot)$.

4.5.2.2 Application to weakly-convex strongly-concave min-max problems

The same technique can be applied to solving weakly-convex strongly-concave min-max problems $\min_{\mathbf{w}} \max_{\mathbf{u}\in\mathcal{U}} f(\mathbf{w},\mathbf{u})$ with a single loop algorithm. In subsection 4.5.1, we assume the partial gradient $\nabla_1 f(\mathbf{w},\mathbf{u})$ is Lipschitz continuous. We replace this assumption by an assumption that $f(\mathbf{w},\mathbf{u})$ is ρ-weakly convex in terms of $\mathbf{w}$ for any $\mathbf{u}\in\mathcal{U}$.

In this case, $F(\mathbf{w}) = \max_{\mathbf{u}\in\mathcal{U}} f(\mathbf{w},\mathbf{u})$ is not smooth but weakly convex. Let us consider its Moreau envelope:

$$\min_{\mathbf{w}} F_\rho(\mathbf{w}) := \min_{\mathbf{u}_1} F(\mathbf{u}_1) + \rho\|\mathbf{u}_1 - \mathbf{w}\|_2^2.$$

This problem is equivalent to

$$\min_{\mathbf{w},\mathbf{u}_1} \max_{\mathbf{u}_2\in\mathcal{U}} f(\mathbf{u}_1,\mathbf{u}_2) + \rho\|\mathbf{u}_1 - \mathbf{w}\|_2^2,$$

which is strongly convex in terms of $\mathbf{u}_1$ and strongly concave in terms of $\mathbf{u}_2$.

Compared to (4.57), this problem just adds another layer of inner maximization. However, it can be still mapped to the general framework as discussed at the beginning. The gradient of $F_\rho(\mathbf{w})$ is given by $\mathcal{M}(\mathbf{w},\mathbf{u}_1^*(\mathbf{w})) = \rho(\mathbf{w} - \mathbf{u}_1^*(\mathbf{w}))$. If we track $\mathbf{u}_1^*(\mathbf{w}_t)$ by $\mathbf{u}_{1,t}$ and its update relies on the gradient $\partial_1 f(\mathbf{u}_{1,t}, \mathbf{u}_2^*(\mathbf{u}_{1,t}))$. Hence, we just need another variable $\mathbf{u}_{2,t}$ to track $\mathbf{u}_2^*(\mathbf{u}_{1,t})$.

We can develop a similar algorithm. First, let us update $\mathbf{u}_1, \mathbf{u}_2$. Given $\mathbf{w}_t, \mathbf{u}_{1,t}, \mathbf{u}_{2,t}$, we update $\mathbf{u}_{1,t+1}, \mathbf{u}_{2,t+1}$ with SGD update by

$$\mathbf{u}_{2,t+1} = \Pi_{\mathcal{U}}[\mathbf{u}_{2,t} + \gamma_2 \partial_2 f(\mathbf{u}_{1,t}, \mathbf{u}_{2,t}; \zeta_t)] \tag{4.61}$$

$$\mathbf{u}_{1,t+1} = \mathbf{u}_{1,t} - \gamma_1(\partial_1 f(\mathbf{u}_{1,t}, \mathbf{u}_{2,t}; \zeta_t) + 2\rho(\mathbf{u}_{1,t} - \mathbf{w}_t)). \tag{4.62}$$

Then we update $\mathbf{w}_{t+1}$ with GD update by

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta 2\rho(\mathbf{w}_t - \mathbf{u}_{1,t}) = (1 - 2\eta\rho)\mathbf{w}_t + 2\eta\rho\mathbf{u}_{1,t}. \tag{4.63}$$

This algorithm also enjoys a complexity of $O(1/\epsilon^4)$ for finding a nearly ϵ-stationary solution of $F(\mathbf{w})$. We refer the readers to (Hu et al., 2024a) for a convergence analysis of this algorithm.

4.5.2.3 Application to Compositional Optimization

We can apply a similar strategy to a compositional function $F(\mathbf{w}) = f_0(g(\mathbf{w}))$, where f_0 is smooth convex and g is weakly convex. With the conjugate of f_0, we can write

$$\min_{\mathbf{w}} f_0(g(\mathbf{w})) = \min_{\mathbf{w}} \max_{\mathbf{u}_2\in\mathcal{U}} f(\mathbf{w},\mathbf{u}_2) := \mathbf{u}_2^\top g(\mathbf{w}) - f_0^*(\mathbf{u}_2).$$

Since f_0 is smooth, then f_0^* is strongly convex. Then if g is weakly convex and $\mathcal{U}$ is bounded (i.e., f_0 is Lipschitz), then $f(\mathbf{w}, \mathbf{u})$ is weakly convex and strongly concave. Optimizing the Moreau envelope of $f_0(g(\mathbf{w}))$ yields:

$$\min_{\mathbf{w},\mathbf{u}_1} \max_{\mathbf{u}_2 \in \mathcal{U}} \mathbf{u}_2^\top g(\mathbf{u}_1) - f_0^*(\mathbf{u}_2) + \rho\|\mathbf{u}_1 - \mathbf{w}\|_2^2,$$

which is strongly convex in terms of $\mathbf{u}_1$ and strongly concave in terms of $\mathbf{u}_2$. We give an update below:

$$\begin{aligned}
\mathbf{u}_{2,t+1} &= \Pi_{\mathcal{U}}[\mathbf{u}_{2,t} + \gamma_2 g(\mathbf{u}_{1,t}; \zeta_t)] \\
\mathbf{u}_{1,t+1} &= \mathbf{u}_{1,t} - \gamma_1(\partial_1 g(\mathbf{u}_{1,t}; \zeta_t)\mathbf{u}_{2,t} + 2\rho(\mathbf{u}_{1,t} - \mathbf{w}_t)) \\
\mathbf{w}_{t+1} &= \mathbf{w}_t - \eta 2\rho(\mathbf{w}_t - \mathbf{u}_{1,t}) = (1 - 2\eta\rho)\mathbf{w}_t + 2\eta\rho\mathbf{u}_{1,t}.
\end{aligned}$$

Then similar convergence analysis can be developed with a complexity of $O(1/\epsilon^4)$ for finding a nearly ϵ-stationary solution to F.

4.5.3 Non-convex Bilevel Optimization

In this section, we discuss the application of the compositional gradient estimation technique to non-convex bilevel optimization defined by

$$\begin{aligned}
&\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w}, \mathbf{u}^*(\mathbf{w})) \\
&\mathbf{u}^*(\mathbf{w}) = \arg\min_{\mathbf{u}\in\mathbb{R}^{d'}} g(\mathbf{w}, \mathbf{u}),
\end{aligned} \tag{4.64}$$

where g is twice differentiable and μ_g-strongly convex in terms of $\mathbf{u}$. Let $F(\mathbf{w}) = f(\mathbf{w}, \mathbf{u}^*(\mathbf{w}))$. The following lemma states the gradient of the objective $F(\mathbf{w})$.

Lemma 4.24 *We have*

$$\nabla F(\mathbf{w}) = \nabla_1 f(\mathbf{w}, \mathbf{u}^*(\mathbf{w})) - \nabla_{21} g(\mathbf{w}, \mathbf{u}^*(\mathbf{w}))^\top (\nabla_{22} g(\mathbf{w}, \mathbf{u}^*(\mathbf{w})))^{-1} \nabla_2 f(\mathbf{w}, \mathbf{u}^*(\mathbf{w})).$$

Proof. By the optimality condition of $\mathbf{u}^*(\mathbf{w})$, we have

$$\nabla_2 g(\mathbf{w}, \mathbf{u}^*(\mathbf{w})) = 0.$$

By taking derivative on both sides, using the chain rule, and the implicit function theorem, we obtain

$$\nabla_{21} g(\mathbf{w}, \mathbf{u}^*(\mathbf{w})) + \nabla_{22} g(\mathbf{w}, \mathbf{u}^*(\mathbf{w})) \frac{\partial \mathbf{u}^*(\mathbf{w})}{\partial \mathbf{w}} = 0.$$

Hence

$$\frac{\partial \mathbf{u}^*(\mathbf{w})}{\partial \mathbf{w}} = -(\nabla_{22} g(\mathbf{w}, \mathbf{u}^*(\mathbf{w})))^{-1} \nabla_{21} g(\mathbf{w}, \mathbf{u}^*(\mathbf{w})).$$

Thus,

$$\begin{aligned}\nabla F(\mathbf{w}) &= \nabla_1 f(\mathbf{w}, \mathbf{u}^*(\mathbf{w})) + \frac{\partial \mathbf{u}^*(\mathbf{w})}{\partial \mathbf{w}}^\top \nabla_2 f(\mathbf{w}, \mathbf{u}^*(\mathbf{w})) \\ &= \nabla_1 f(\mathbf{w}, \mathbf{u}^*(\mathbf{w})) - \nabla_{21} g(\mathbf{w}, \mathbf{u}^*(\mathbf{w}))^\top (\nabla_{22} g(\mathbf{w}, \mathbf{u}^*(\mathbf{w})))^{-1} \nabla_2 f(\mathbf{w}, \mathbf{u}^*(\mathbf{w})).\end{aligned}$$

□

Let us define

$$\begin{aligned}&\mathcal{M}(\mathbf{w}, \mathbf{u}^*(\mathbf{w})) = \\ &\nabla_1 f(\mathbf{w}, \mathbf{u}^*(\mathbf{w})) - \nabla_{21} g(\mathbf{w}, \mathbf{u}^*(\mathbf{w}))^\top (\nabla_{22} g(\mathbf{w}, \mathbf{u}^*(\mathbf{w})))^{-1} \nabla_2 f(\mathbf{w}, \mathbf{u}^*(\mathbf{w})).\end{aligned}$$

If we can establish the Lipschitz continuity of $\mathcal{M}(\mathbf{w}, \mathbf{u}^*(\mathbf{w}))$ in terms of the second argument and the Lipschitz continuity of $\mathbf{u}^*(\mathbf{w})$, then the similar technique can be leveraged. Let $\mathbf{u}^*(\mathbf{w}_t)$ be tracked by $\mathbf{u}_t$. It can be updated by SGD:

$$\mathbf{u}_{t+1} = \mathbf{u}_t - \gamma_t \nabla_2 g(\mathbf{w}_t, \mathbf{u}_t; \zeta_t). \tag{4.65}$$

With $\mathbf{u}_t$, the gradient at $\mathbf{w}_t$ can be estimated by

$$\mathcal{M}(\mathbf{w}_t, \mathbf{u}_t) = \nabla_1 f(\mathbf{w}_t, \mathbf{u}_t) + \nabla_{21} g(\mathbf{w}_t, \mathbf{u}_t)^\top (\nabla_{22} g(\mathbf{w}_t, \mathbf{u}_t))^{-1} \nabla_2 f(\mathbf{w}_t, \mathbf{u}_t). \tag{4.66}$$

However, another challenge is to handle the Hessian inverse $(\nabla_{22} g(\mathbf{w}_t, \mathbf{u}_t)^{-1}$, which itself is a compositional structure. We will discuss three different ways to tackle this challenge. If we have a stochastic estimator of $\mathcal{M}(\mathbf{w}_t, \mathbf{u}_t)$ denoted by $\mathbf{v}_t$, then we update the model parameter by:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{v}_t. \tag{4.67}$$

4.5.3.1 Approach 1: The MA Estimator

If the lower level problem is low-dimensional such that the inverse of the Hessian matrix can be efficiently computed, we can estimate $\nabla_{22} g(\mathbf{w}_t, \mathbf{u}_t)$ by a MA estimator:

$$H_{22,t} = S_{\mu_g}[(1-\beta) H_{22,t-1} + \beta \nabla_{22} g(\mathbf{w}_t, \mathbf{u}_t; \zeta_{2,t})].$$

where $S_{\mu_g}[\cdot]$ is a projection operator that projects a matrix into a matrix whose minimum eigen-value is lower bounded by μ_g, where μ_g is the lower bound of eigen-values of $\nabla_{22} g(\mathbf{w}, \mathbf{u})$. The projection ensures that $[H_{22,t}]^{-1}$ is Lipschitz continuous with respect to $H_{22,t}$.

The a vanilla stochastic gradient estimator of $\mathbf{w}_t$ and its MA estimator are computed by

$$\begin{aligned}\mathbf{z}_t &= \nabla_1 f(\mathbf{w}_t, \mathbf{u}_t; \xi_t) + \nabla_{21} g(\mathbf{w}_t, \mathbf{u}_t; \zeta'_{2,t})^\top (H_{22,t})^{-1} \nabla_2 f(\mathbf{w}_t, \mathbf{u}_t; \xi_t) \\ \mathbf{v}_t &= (1-\beta)\mathbf{v}_{t-1} + \beta \mathbf{z}_t.\end{aligned} \tag{4.68}$$

Convergence Analysis

The proof is largely similar to that of Theorem 4.3. We provide a sketch of proof below. Recall that

$$\mathcal{M}(\mathbf{w}_t, \mathbf{u}_t) = \nabla_1 f(\mathbf{w}_t, \mathbf{u}_t) + \nabla_{21} g(\mathbf{w}_t, \mathbf{u}_t)^\top (\nabla_{22} g(\mathbf{w}_t, \mathbf{u}_t))^{-1} \nabla_2 f(\mathbf{w}_t, \mathbf{u}_t).$$

Define:

$$\hat{\mathcal{M}}(\mathbf{w}_t, \mathbf{u}_t) = \nabla_1 f(\mathbf{w}_t, \mathbf{u}_t) + \nabla_{21} g(\mathbf{w}_t, \mathbf{u}_t)^\top H_{22,t}^{-1} \nabla_2 f(\mathbf{w}_t, \mathbf{u}_t).$$

First, similar to Lemma 4.9, we have the following:

$$F(\mathbf{w}_{t+1}) \leq F(\mathbf{w}_t) + \frac{\eta_t}{2}\|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|_2^2 - \frac{\eta_t}{2}\|\nabla F(\mathbf{w}_t)\|_2^2 - \frac{1}{4\eta_t}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2. \tag{4.69}$$

We establish a recursion of the error $\|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|_2^2$ similar to Lemma 4.7 by noting that $\mathbb{E}_{\xi_t, \zeta'_{2,t}}[\mathbf{z}_t] = \hat{\mathcal{M}}(\mathbf{w}_t, \mathbf{u}_t)$ and there exists $\sigma > 0$ such that $\mathbb{E}_{\xi_t, \zeta'_{2,t}}[\|\mathbf{z}_t - \hat{\mathcal{M}}(\mathbf{w}_t, \mathbf{u}_t)\|_2^2] \leq \sigma^2$. Thus, Lemma 4.7 implies that

$$\begin{aligned}\mathbb{E}_{\xi_t, \zeta'_{2,t}}\left[\|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|_2^2\right] &\leq (1-\beta_t)\|\mathbf{v}_{t-1} - \nabla F(\mathbf{w}_{t-1})\|_2^2 \\ &+ \frac{2L_F^2}{\beta_t}\|\mathbf{w}_{t-1} - \mathbf{w}_t\|_2^2 + 4\beta_t \left\|\hat{\mathcal{M}}(\mathbf{w}_t, \mathbf{u}_t) - \nabla F(\mathbf{w}_t)\right\|_2^2 + \beta_t^2 \sigma^2.\end{aligned} \tag{4.70}$$

Then, we bound $\|\hat{\mathcal{M}}(\mathbf{w}_t, \mathbf{u}_t) - \nabla F(\mathbf{w}_t)\|_2^2$ by

$$\begin{aligned}\|\hat{\mathcal{M}}(\mathbf{w}_t, \mathbf{u}_t) - \nabla F(\mathbf{w}_t)\|_2^2 &\leq 2\|\hat{\mathcal{M}}(\mathbf{w}_t, \mathbf{u}_t) - \mathcal{M}(\mathbf{w}_t, \mathbf{u}_t)\|_2^2 \\ &+ 2\|\mathcal{M}(\mathbf{w}_t, \mathbf{u}_t) - \nabla F(\mathbf{w}_t)\|_2^2 \\ &\leq O(\|H_{22,t} - \nabla_{22} g(\mathbf{w}_t, \mathbf{u}_t)\|_2^2) + O(\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\|_2^2).\end{aligned}$$

As a result, we have

$$\begin{aligned}\mathbb{E}[\|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|_2^2] &\leq (1-\beta_t)\|\mathbf{v}_{t-1} - \nabla F(\mathbf{w}_{t-1})\|^2 + \frac{2L_F^2}{\beta_t}\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2 \\ &+ \beta_t(O(\|H_{22,t} - \nabla_{22} g(\mathbf{w}_t, \mathbf{u}_t)\|_2^2) + O(\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\|_2^2)) + \beta_t^2 O(\sigma^2).\end{aligned}$$

This result is similar to that in Lemma 4.8.

We can further build the error recursion of $\|H_{22,t} - \nabla_{22} g(\mathbf{w}_t, \mathbf{u}_t)\|_2^2$ similar to Lemma 4.1, and the error recursion of $\|\mathbf{u}_t - \mathbf{u}^*(\mathbf{w}_t)\|_2^2$ similar to Lemma 4.20.

Combining these results, we can establish a complexity of $O(1/\epsilon^4)$ for finding an ϵ-stationary solution of $F(\cdot)$ in expectation.

4.5.3.2 Approach 2: The Neumann Series (Matrix Taylor Approximation)

If the lower level problem is high-dimensional such that it is prohibited to compute the Hessian, one approach is to leverage the Neuman series:

$$A^{-1} = \sum_{i=0}^{\infty} (I - A)^i, \quad \text{if} \quad \|A\| \leq 1. \tag{4.71}$$

Hence, if $\|\nabla_{22} g(\mathbf{w}_t, \mathbf{u}_t)\| \leq L_{22}$, we estimate the inverse of $\frac{1}{L_{22}} \nabla_{22} g(\mathbf{w}_t, \mathbf{u}_t)$, yielding

$$(\nabla_{22} g(\mathbf{w}_t, \mathbf{u}_t))^{-1} \approx \frac{1}{L_{22}} \sum_{i=0}^{K-1} \left(I - \frac{1}{L_{22}} \nabla_{22} g(\mathbf{w}_t, \mathbf{u}_t) \right)^i. \tag{4.72}$$

This can be further estimated by a stochastic route, by sampling k from $\{0, \ldots, K-1\}$ randomly, then estimate the Hessian inverse by

$$Q_{22,t} = \begin{cases} \frac{K}{L_{22}} \prod_{i=1}^{k} \left(I - \frac{1}{L_{22}} \nabla_{22} g(\mathbf{w}_t, \mathbf{u}_t; \zeta_i) \right) & \text{if } k \geq 1 \\ \frac{K}{L_{22}} I & \text{if } k = 0 \end{cases}. \tag{4.73}$$

This is can be justified by

$$\begin{aligned} \mathbb{E}[Q_{22,t}] &= \frac{1}{K} \frac{K}{L_{22}} I + \frac{K-1}{K} \mathbb{E}_{k \sim \{1,\ldots,K-1\}} \left[\frac{K}{L_{22}} \prod_{i=1}^{k} \left(I - \frac{1}{L_{22}} \mathbb{E}[\nabla_{22} g(\mathbf{w}_t, \mathbf{u}_t; \zeta_i)] \right) \right] \\ &= \mathbb{E}_k \frac{K}{L_{22}} \left(I - \frac{1}{L_2} \nabla_{22} g(\mathbf{w}_t, \mathbf{u}_t) \right)^k = \sum_{k=0}^{K-1} \frac{1}{L_{22}} \left(I - \frac{1}{L_{22}} \nabla_{22} g(\mathbf{w}_t, \mathbf{u}_t) \right)^k. \end{aligned}$$

Then the vanilla gradient estimator of $\mathbf{w}_t$ and its MA estimator are computed by

$$\begin{aligned} \mathbf{z}_t &= \nabla_1 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_{1,t}) + \nabla_{21} g(\mathbf{w}_t, \mathbf{u}_t; \zeta'_{2,t})^\top Q_{22,t} \nabla_2 f(\mathbf{w}_t, \mathbf{u}_t; \zeta_{1,t}) \\ \mathbf{v}_t &= (1 - \beta) \mathbf{v}_{t-1} + \beta \mathbf{z}_t. \end{aligned} \tag{4.74}$$

Convergence Analysis

We provide a proof sketch below. We can understand that $\mathbf{z}_t$ is a unbiased stochastic estimator of

$$\hat{\mathcal{M}}(\mathbf{w}_t, \mathbf{u}_t) = \nabla_1 f(\mathbf{w}_t, \mathbf{u}_t) + \nabla_{21} g(\mathbf{w}_t, \mathbf{u}_t)^\top \mathbb{E}[Q_{22,t}] \nabla_2 f(\mathbf{w}_t, \mathbf{u}_t).$$

We decompose the estimation error of $\mathbf{v}_t$ similarly as in (4.70) and bound $\|\hat{\mathcal{M}}(\mathbf{w}_t, \mathbf{u}_t) - \nabla F(\mathbf{w}_t)\|_2^2$ by

$$\begin{aligned}\|\hat{\mathcal{M}}(\mathbf{w}_t, \mathbf{u}_t) - \nabla F(\mathbf{w}_t)\|_2^2 \leq\ & 2\|\mathcal{M}(\mathbf{w}_t, \mathbf{u}_t) - \nabla F(\mathbf{w}_t)\|_2^2 \\ & + 2\|\hat{\mathcal{M}}(\mathbf{w}_t, \mathbf{u}_t) - \mathcal{M}(\mathbf{w}_t, \mathbf{u}_t)\|_2^2.\end{aligned}$$

The error recursion of the first term on the right hand side can be similarly bounded as before. To bound the last error, since

$$\left[\nabla_{22}^2 g(\mathbf{w}, \mathbf{u})\right]^{-1} = \mathbb{E}[Q_{22}] + \frac{1}{L_{22}} \sum_{i=K}^{\infty} \left[I - \frac{1}{L_{22}} \nabla_{22}^2 g(\mathbf{w}, \mathbf{u})\right]^i,$$

we have

$$\begin{aligned}&\|\mathcal{M}(\mathbf{w}_t, \mathbf{u}_t) - \hat{\mathcal{M}}(\mathbf{w}_t, \mathbf{u}_t)\|_2^2 \leq O(\| \left[\nabla_{22}^2 g(\mathbf{w}, \mathbf{u})\right]^{-1} - \mathbb{E}[Q_{22}]\|_2^2) \\ &\left\| \left[\nabla_{22}^2 g(\mathbf{w}, \mathbf{u})\right]^{-1} - \mathbb{E}[Q_{22}] \right\|_2 \leq \frac{1}{L_{22}} \sum_{i=K}^{\infty} \left\| I - \frac{1}{L_{22}} \nabla_{22}^2 g(\mathbf{w}, \mathbf{u}) \right\|_2^i \leq \frac{1}{\mu_g} \left(1 - \frac{\mu_g}{L_{22}}\right)^K.\end{aligned}$$

As a result, if $K = O(\frac{L_{22}}{\mu_g} \log(1/(\mu_g \beta_t \sigma^2)))$, then $\|\mathcal{M}(\mathbf{w}_t, \mathbf{u}_t) - \mathcal{M}'(\mathbf{w}_t, \mathbf{u}_t)\|_2^2 \leq O(\beta_t \sigma^2)$. Then similar to the analysis of approach 1, we can establish a complexity of $O(1/\epsilon^4)$ for finding an ϵ-stationary solution of $F(\cdot)$ in expectation.

4.5.3.3 Approach 3: The penalty method

An alternative approach to avoid computing the Hessian inverse and Jacobian matrices is to reformulate the problem as a constrained optimization problem:

$$\begin{aligned}\min_{\mathbf{w},\mathbf{u}} \quad & f(\mathbf{w}, \mathbf{u}) \\ \text{s.t.} \quad & g(\mathbf{w}, \mathbf{u}) \leq \min_{\mathbf{u}} g(\mathbf{w}, \mathbf{u}).\end{aligned}$$

This constrained problem can be addressed using a penalty method (see Chapter 6.7):

$$\min_{\mathbf{w},\mathbf{u}} f(\mathbf{w}, \mathbf{u}) + \lambda \big(g(\mathbf{w}, \mathbf{u}) - \min_{\mathbf{y}} g(\mathbf{w}, \mathbf{y})\big)_+,$$

where $\lambda > 0$ is a penalty parameter and $(\cdot)_+$ denotes the positive part. Since $g(\mathbf{w}, \mathbf{u}) \geq \min_{\mathbf{y}} g(\mathbf{w}, \mathbf{y})$, the formulation simplifies to:

$$\min_{\mathbf{w},\mathbf{u}} f(\mathbf{w}, \mathbf{u}) + \lambda \left(g(\mathbf{w}, \mathbf{u}) - \min_{\mathbf{y}} g(\mathbf{w}, \mathbf{y})\right) \tag{4.75}$$

$$= \min_{\mathbf{w},\mathbf{u}} \max_{\mathbf{y}} f(\mathbf{w}, \mathbf{u}) + \lambda \left(g(\mathbf{w}, \mathbf{u}) - g(\mathbf{w}, \mathbf{y})\right). \tag{4.76}$$

If both f and g are smooth and g is strongly convex in its second argument, the resulting formulation becomes a *non-convex strongly-concave min-max problem*, which can be effectively addressed using the SMDA algorithm with the following update for $t \geq 1$:

$$\begin{aligned}
&\mathbf{y}_{t+1} = \mathbf{y}_t + \gamma_t \lambda \nabla_2 g(\mathbf{w}_t, \mathbf{y}_t; \xi_t), \\
&\mathbf{z}_t = \nabla f(\mathbf{w}_t, \mathbf{u}_t; \zeta_t) + \lambda \left(\nabla g(\mathbf{w}_t, \mathbf{u}_t; \xi_t) - \begin{bmatrix} \nabla_1 g(\mathbf{w}_t, \mathbf{y}_t; \xi_t) \\ 0 \end{bmatrix} \right), \\
&\mathbf{v}_t = (1 - \beta_t)\mathbf{v}_{t-1} + \beta_t \mathbf{z}_t, \\
&\begin{bmatrix} \mathbf{w}_{t+1} \\ \mathbf{u}_{t+1} \end{bmatrix} = \begin{bmatrix} \mathbf{w}_t \\ \mathbf{u}_t \end{bmatrix} - \eta_t \mathbf{v}_t.
\end{aligned} \tag{4.77}$$

Convergence Analysis

The convergence analysis of (4.77) for the min–max problem (4.75) follows a similar approach to that of Theorem 4.5 for SMDA. However, a remaining challenge lies in converting the convergence result for the min–max formulation into that of the original problem. To address this, we provide the detailed convergence analysis below. We begin by stating the following assumption.

Assumption 4.9. *Regarding the problem (4.64), the following conditions hold:*

(i) $g(\mathbf{w}, \mathbf{u})$ *is* μ*-strongly concave in terms of* $\mathbf{u}$.
(ii) $\nabla f(\mathbf{w}, \mathbf{u})$ *is* L_f*-Lipschitz continuous such that*

$$\|\nabla f(\mathbf{w}_1, \mathbf{u}_1) - \nabla f(\mathbf{w}_2, \mathbf{u}_2)\|_2 \leq L_f \left\| \begin{pmatrix} \mathbf{w}_1 \\ \mathbf{u}_1 \end{pmatrix} - \begin{pmatrix} \mathbf{w}_2 \\ \mathbf{u}_2 \end{pmatrix} \right\|_2. \tag{4.78}$$

(iii) $\nabla g(\mathbf{w}, \mathbf{u})$ *is* L_g*-Lipschitz continuous such that*

$$\|\nabla g(\mathbf{w}_1, \mathbf{u}_1) - \nabla g(\mathbf{w}_2, \mathbf{u}_2)\|_2 \leq L_g \left\| \begin{pmatrix} \mathbf{w}_1 \\ \mathbf{u}_1 \end{pmatrix} - \begin{pmatrix} \mathbf{w}_2 \\ \mathbf{u}_2 \end{pmatrix} \right\|_2. \tag{4.79}$$

(iv) there exist σ_f, σ_g *such that*

$$\mathbb{E}[\|\nabla f(\mathbf{w}, \mathbf{u}; \zeta) - \nabla f(\mathbf{w}, \mathbf{u})\|_2^2] \leq \sigma_f^2, \tag{4.80}$$

$$\mathbb{E}[\|\nabla g(\mathbf{w}, \mathbf{u}; \xi) - \nabla g(\mathbf{w}, \mathbf{u})\|_2^2] \leq \sigma_g^2. \tag{4.81}$$

(v) $\min_{\mathbf{w},\mathbf{u}} f(\mathbf{w}, \mathbf{u}) \geq -\infty$.

Let us define $\bar{\mathbf{w}} = (\mathbf{w}, \mathbf{u})$ and

$$\bar{f}(\bar{\mathbf{w}}, \mathbf{y}) = f(\mathbf{w}, \mathbf{u}) + \lambda \left(g(\mathbf{w}, \mathbf{u}) - g(\mathbf{w}, \mathbf{y}) \right) \tag{4.82}$$

$$\bar{F}(\bar{\mathbf{w}}) = \max_{\mathbf{y}} \bar{f}(\bar{\mathbf{w}}, \mathbf{y}). \tag{4.83}$$

Then

$$\begin{aligned}
\nabla_1 \bar{f}(\bar{\mathbf{w}}, \mathbf{y}) &= \nabla f(\mathbf{w}, \mathbf{u}) + \lambda \left(\nabla g(\mathbf{w}, \mathbf{u}) - \begin{bmatrix} \nabla_1 g(\mathbf{w}, \mathbf{y}) \\ 0 \end{bmatrix} \right), \\
\nabla_2 \bar{f}(\bar{\mathbf{w}}, \mathbf{y}) &= -\lambda \nabla_2 g(\mathbf{w}, \mathbf{y}), \\
\nabla_1 \bar{f}(\bar{\mathbf{w}}, \mathbf{y}; \varepsilon) &= \nabla f(\mathbf{w}, \mathbf{u}; \zeta) + \lambda \left(\nabla g(\mathbf{w}, \mathbf{u}; \xi) - \begin{bmatrix} \nabla_1 g(\mathbf{w}, \mathbf{y}; \xi) \\ 0 \end{bmatrix} \right), \\
\nabla_2 \bar{f}(\bar{\mathbf{w}}, \mathbf{y}; \xi) &= -\lambda \nabla_2 g(\mathbf{w}, \mathbf{y}; \xi).
\end{aligned}$$

where $\varepsilon = (\zeta, \xi)$. We first show $\bar{f}(\bar{\mathbf{w}}, \mathbf{y})$ satisfies the conditions in Assumption (4.8).

Lemma 4.25 *Under Assumption 4.9, we have*

(i) $\bar{f}(\bar{\mathbf{w}}, \mathbf{y})$ *is* $\mu\lambda$*-strongly concave in terms of* $\mathbf{u}$.
(ii) $\nabla_1 \bar{f}(\bar{\mathbf{w}}, \mathbf{y})$ *is Lipschitz continuous, i.e.,*

$$\|\nabla_1 \bar{f}(\bar{\mathbf{w}}_1, \mathbf{y}_1) - \nabla_1 \bar{f}(\bar{\mathbf{w}}_2, \mathbf{y}_2)\|_2 \le (L_f + 2L_g\lambda)(\|\bar{\mathbf{w}}_1 - \bar{\mathbf{w}}_2\|_2 + \|\mathbf{y}_1 - \mathbf{y}_2\|_2).$$

(iii) $\nabla_2 \bar{f}(\bar{\mathbf{w}}, \mathbf{y})$ *is Lipschitz continuous, i.e.,*

$$\|\nabla_2 \bar{f}(\bar{\mathbf{w}}_1, \mathbf{y}_1) - \nabla_2 f(\bar{\mathbf{w}}_2, \mathbf{y}_2)\|_2 \le L_g\lambda\|\bar{\mathbf{w}}_1 - \bar{\mathbf{w}}_2\|_2 + L_g\lambda\|\mathbf{y}_1 - \mathbf{y}_2\|_2.$$

(iv)

$$\begin{aligned}
\mathbb{E}[\|\nabla_1 \bar{f}(\bar{\mathbf{w}}, \mathbf{y}; \varepsilon) - \nabla_1 \bar{f}(\bar{\mathbf{w}}, \mathbf{y})\|_2^2] &\le 3\sigma_f^2 + 6\lambda^2\sigma_g^2, \\
\mathbb{E}[\|\nabla_2 \bar{f}(\bar{\mathbf{w}}, \mathbf{y}; \xi) - \nabla_2 f(\bar{\mathbf{w}}, \mathbf{y})\|_2^2] &\le \lambda^2\sigma_g^2.
\end{aligned}$$

(v) $\bar{F}(\bar{\mathbf{w}}) := \max_{\mathbf{y}} \bar{f}(\bar{\mathbf{w}}, \mathbf{y}) \ge -\infty$.

Proof. (i) is obvious. The Lipschitz continuity of $\nabla_1 \bar{f}(\bar{\mathbf{w}}, \mathbf{y})$ follows that of $\nabla f(\mathbf{w}, \mathbf{u})$ and $\nabla g(\mathbf{w}, \mathbf{u})$. For (iii), we have

$$\begin{aligned}
&\|\nabla_2 \bar{f}(\bar{\mathbf{w}}_1, \mathbf{y}_1) - \nabla_2 \bar{f}(\bar{\mathbf{w}}_2, \mathbf{y}_2)\|_2 = \lambda\|\nabla_2 g(\mathbf{w}_1, \mathbf{u}_1) - \nabla_2 g(\mathbf{w}_2, \mathbf{u}_2)\|_2 \\
&\le \lambda\|\nabla g(\mathbf{w}_1, \mathbf{u}_1) - \nabla g(\mathbf{w}_2, \mathbf{u}_2)\|_2 \le \lambda L_g \left\| \begin{pmatrix} \mathbf{w}_1 \\ \mathbf{u}_1 \end{pmatrix} - \begin{pmatrix} \mathbf{w}_2 \\ \mathbf{u}_2 \end{pmatrix} \right\|_2 \\
&\le \lambda L_g(\|\mathbf{w}_1 - \mathbf{w}_2\|_2 + \|\mathbf{u}_1 - \mathbf{u}_2\|_2) \le \lambda L_g(\|\bar{\mathbf{w}}_1 - \bar{\mathbf{w}}_2\|_2 + \|\mathbf{u}_1 - \mathbf{u}_2\|_2).
\end{aligned}$$

It is trivial to prove (iv). The last result follows that $\max_{\mathbf{y}} \bar{f}(\bar{\mathbf{w}}, \mathbf{y}) \ge f(\mathbf{w}, \mathbf{u}) \ge \infty$. □

Theorem 4.6 *Suppose Assumption 4.9 hold. By setting*

$$\beta_t = \beta = \frac{\epsilon^2}{9\sigma_f^2 + 18\lambda^2\sigma_g^2},$$

$$\gamma_t = \gamma = \frac{\mu_g \epsilon^2}{96(L_f + 2L_g\lambda)^2\lambda\sigma_g^2},$$

$$\eta_t = \min\left\{\frac{\beta}{\sqrt{8}(L_f + 2L_g\lambda)(1+L_g)}, \frac{\gamma\mu_g\lambda}{16\sqrt{3}(L_f + 2L_g\lambda)L_g}, \frac{1}{2(L_f + 2L_g\lambda)(1+L_g)}\right\}$$

in (4.77), then the following holds

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1}\left\{\frac{1}{4}\|\mathbf{v}_t\|_2^2 + \left\|\nabla\bar{F}(\bar{\mathbf{w}}_t)\right\|_2^2\right\}\right] \leq \epsilon^2, \tag{4.84}$$

with an iteration complexity of

$$T = O\left(\max\left\{\frac{C_\Upsilon\lambda}{\epsilon^2}, \frac{C_\Upsilon(\lambda\sigma_f^2 + \lambda^3\sigma_g^2)}{\epsilon^4}, \frac{C_\Upsilon\lambda^3\sigma_g^2}{\epsilon^4\mu_g^2}\right\}\right), \tag{4.85}$$

where $C_\Upsilon = 2(\bar{F}(\bar{\mathbf{w}}_0) - \min_{\bar{\mathbf{w}}}\bar{F}(\bar{\mathbf{w}})) + \frac{1}{\sqrt{8}L_F}\|\mathbf{v}_0 - \nabla\bar{F}(\bar{\mathbf{w}}_0)\|_2^2 + \frac{L_1}{\sqrt{3}\kappa}\|\mathbf{y}_0 - \mathbf{y}^*(\mathbf{w}_0)\|_2^2$.

Proof. We map the problem into the setting in Theorem 4.5 with $L_1 = L_f + 2L_g\lambda, L_{21} = L_g\lambda, L_2 = L_g\lambda, \mu = \mu_g\lambda, \kappa = L_{21}/(\mu_g\lambda) = L_g, L_F = L_1(1+\kappa) = (L_f + 2L_g\lambda)(1+L_g), \sigma_1^2 = 3\sigma_f^2 + 6\lambda^2\sigma_g^2, \sigma_2^2 = \lambda^2\sigma_g^2$. Then, plugging these values into the result in Theorem 4.5, we obtain the results. □

Convergence of the original function

Next, we derive the convergence of the original function in terms of $\|\nabla F(\mathbf{w})\|_2$. We need the following additional assumption.

Assumption 4.10. *(i) g is twice differentiable and $\nabla_{21}g(\mathbf{w},\mathbf{u})$ and $\nabla g_{22}(\mathbf{w},\mathbf{u})$ are L_{gg}-Lipschitz continuous; and (ii) $\|\nabla_2 f(\mathbf{w},\mathbf{u})\|_2 \leq G_f$.*

Lemma 4.26 *Let $\mathbf{u}_\lambda^*(\mathbf{w}) = \arg\min_{\mathbf{u}}\bar{F}(\mathbf{w},\mathbf{u}), \mathbf{u}^*(\mathbf{w}) = \arg\min_{\mathbf{u}} g(\mathbf{w},\mathbf{u})$. Under Assumption 4.10(i), we have*

$$\|\nabla F(\mathbf{w}) - \nabla_1\bar{F}(\mathbf{w},\mathbf{u}_\lambda^*(\mathbf{w}))\|_2 \leq L_f(1 + \frac{L_g}{\mu_g})\|\mathbf{u}_\lambda^*(\mathbf{w}) - \mathbf{u}^*(\mathbf{w})\|_2 + L_{gg}\lambda(1 + \frac{L_g}{\mu_g})\|\mathbf{u}_\lambda^*(\mathbf{w}) - \mathbf{u}^*(\mathbf{w})\|_2^2.$$

Proof. Let $\mathbf{u}^* = \mathbf{u}^*(\mathbf{w})$. Then,

$$\nabla_1 \bar{F}(\mathbf{w}, \mathbf{u}) = \nabla_1 f(\mathbf{w}, \mathbf{u}) + \lambda(\nabla_1 g(\mathbf{w}, \mathbf{u}) - \nabla_1 g(\mathbf{w}, \mathbf{u}^*))$$
$$\nabla_2 \bar{F}(\mathbf{w}, \mathbf{u}) = \nabla_2 f(\mathbf{w}, \mathbf{u}) + \lambda \nabla_2 g(\mathbf{w}, \mathbf{u}).$$

Due to Lemma 4.24, we have

$$\begin{aligned}\nabla F(\mathbf{w}) - \nabla_1 \bar{F}(\mathbf{w}, \mathbf{u}) &= \nabla_1 f(\mathbf{w}, \mathbf{u}^*) - \nabla_1 f(\mathbf{w}, \mathbf{u}) \\ &- \nabla_{12} g(\mathbf{w}, \mathbf{u}^*) \nabla_{22} g(\mathbf{w}, \mathbf{u}^*)^{-1} \nabla_2 f(\mathbf{w}, \mathbf{u}^*) - \lambda(\nabla_1 g(\mathbf{w}, \mathbf{u}) - \nabla_1 g(\mathbf{w}, \mathbf{u}^*)).\end{aligned} \tag{4.86}$$

We can rearrange terms for $(\nabla_1 g(\mathbf{w}, \mathbf{u}) - \nabla_1 g(\mathbf{w}, \mathbf{u}^*))$ as the following:

$$\begin{aligned}\nabla_1 g(\mathbf{w}, \mathbf{u}) - \nabla_1 g(\mathbf{w}, \mathbf{u}^*) = \nabla_1 g(\mathbf{w}, \mathbf{u}) - \nabla_1 g(\mathbf{w}, \mathbf{u}^*) - \nabla_{12} g(\mathbf{w}, \mathbf{u}^*)^\top (\mathbf{u} - \mathbf{u}^*) \\ + \nabla_{12} g(\mathbf{w}, \mathbf{u}^*)^\top (\mathbf{u} - \mathbf{u}^*).\end{aligned} \tag{4.87}$$

To continue, we have

$$\begin{aligned}\mathbf{u} - \mathbf{u}^* = &- \nabla_{22} g(\mathbf{w}, \mathbf{u}^*)^{-1} (\nabla_2 g(\mathbf{w}, \mathbf{u}) - \nabla_2 g(\mathbf{w}, \mathbf{u}^*) - \nabla_{22} g(\mathbf{w}, \mathbf{u}^*)(\mathbf{u} - \mathbf{u}^*)) \\ &+ \nabla_{22} g(\mathbf{w}, \mathbf{u}^*)^{-1} (\nabla_2 g(\mathbf{w}, \mathbf{u}) - \nabla_2 g(\mathbf{w}, \mathbf{u}^*)).\end{aligned}$$

By the optimality condition for $\mathbf{u}^*$, $\nabla_2 g(\mathbf{w}, \mathbf{u}^*) = 0$, and $\nabla_2 \bar{F}(\mathbf{w}, \mathbf{u}) = \nabla_2 f(\mathbf{w}, \mathbf{u}) + \lambda \nabla_2 g(\mathbf{w}, \mathbf{u})$, we can express $\mathbf{u} - \mathbf{u}^*$ as

$$\begin{aligned}\mathbf{u} - \mathbf{u}^* = &- \nabla_{22} g(\mathbf{w}, \mathbf{u}^*)^{-1} (\nabla_2 g(\mathbf{w}, \mathbf{u}) - \nabla_2 g(\mathbf{w}, \mathbf{u}^*) - \nabla_{22} g(\mathbf{w}, \mathbf{u}^*)(\mathbf{u} - \mathbf{u}^*)) \\ &+ \frac{1}{\lambda} \nabla_{22} g(\mathbf{w}, \mathbf{u}^*)^{-1} (\nabla_2 \bar{F}(\mathbf{w}, \mathbf{u}) - \nabla_2 f(\mathbf{w}, \mathbf{u})).\end{aligned} \tag{4.88}$$

Plugging (4.87) and (4.88) back to (4.86), we have

$$\begin{aligned}&\nabla F(\mathbf{w}) - \nabla_1 \bar{F}(\mathbf{w}, \mathbf{u}) = \nabla_1 f(\mathbf{w}, \mathbf{u}^*) - \nabla_1 f(\mathbf{w}, \mathbf{u}) \\ &- \nabla_{12} g(\mathbf{w}, \mathbf{u}^*) \nabla_{22} g(\mathbf{w}, \mathbf{u}^*)^{-1} \nabla_2 f(\mathbf{w}, \mathbf{u}^*) \\ &- \lambda(\nabla_1 g(\mathbf{w}, \mathbf{u}) - \nabla_1 g(\mathbf{w}, \mathbf{u}^*) - \nabla_{12} g(\mathbf{w}, \mathbf{u}^*)^\top (\mathbf{u} - \mathbf{u}^*)) \\ &+ \lambda \nabla_{12} g(\mathbf{w}, \mathbf{u}^*)^\top \nabla_{22} g(\mathbf{w}, \mathbf{u}^*)^{-1} (\nabla_2 g(\mathbf{w}, \mathbf{u}) - \nabla_2 g(\mathbf{w}, \mathbf{u}^*) - \nabla_{22} g(\mathbf{w}, \mathbf{u}^*)(\mathbf{u} - \mathbf{u}^*)) \\ &- \nabla_{12} g(\mathbf{w}, \mathbf{u}^*)^\top \nabla_{22} g(\mathbf{w}, \mathbf{u}^*)^{-1} (\nabla_2 \bar{F}(\mathbf{w}, \mathbf{u}) - \nabla_2 f(\mathbf{w}, \mathbf{u})).\end{aligned}$$

As a result, we have

$$\begin{aligned}&\nabla F(\mathbf{w}) - \nabla_1 \bar{F}(\mathbf{w}, \mathbf{u}) + \nabla_{12} g(\mathbf{w}, \mathbf{u}^*)^\top \nabla_{22} g(\mathbf{w}, \mathbf{u}^*)^{-1} \nabla_2 \bar{F}(\mathbf{w}, \mathbf{u}) \\ &= \nabla_1 f(\mathbf{w}, \mathbf{u}^*) - \nabla_1 f(\mathbf{w}, \mathbf{u}) \\ &- \nabla_{12} g(\mathbf{w}, \mathbf{u}^*) \nabla_{22} g(\mathbf{w}, \mathbf{u}^*)^{-1} (\nabla_2 f(\mathbf{w}, \mathbf{u}^*) - \nabla_2 f(\mathbf{w}, \mathbf{u})) \\ &- \lambda(\nabla_1 g(\mathbf{w}, \mathbf{u}) - \nabla_1 g(\mathbf{w}, \mathbf{u}^*) - \nabla_{12} g(\mathbf{w}, \mathbf{u}^*)^\top (\mathbf{u} - \mathbf{u}^*)) \\ &+ \lambda \nabla_{12} g(\mathbf{w}, \mathbf{u}^*)^\top \nabla_{22} g(\mathbf{w}, \mathbf{u}^*)^{-1} (\nabla_2 g(\mathbf{w}, \mathbf{u}) - \nabla_2 g(\mathbf{w}, \mathbf{u}^*) - \nabla_{22} g(\mathbf{w}, \mathbf{u}^*)(\mathbf{u} - \mathbf{u}^*)).\end{aligned}$$

By the Assumption 4.10 we have

$$\|\nabla_1 g(\mathbf{w},\mathbf{u}) - \nabla_1 g(\mathbf{w},\mathbf{u}^*) - \nabla_{12} g(\mathbf{w},\mathbf{u}^*)^\top(\mathbf{u}-\mathbf{u}^*)\|_2 \leq L_{gg}\|\mathbf{u}-\mathbf{u}^*\|_2^2,$$
$$\|\nabla_2 g(\mathbf{w},\mathbf{u}) - \nabla_2 g(\mathbf{w},\mathbf{u}^*) - \nabla_{22} g(\mathbf{w},\mathbf{u}^*)(\mathbf{u}-\mathbf{u}^*)\|_2 \leq L_{gg}\|\mathbf{u}-\mathbf{u}^*\|_2^2.$$

By the Assumption 4.9 we have

$$\|\nabla_1 f(\mathbf{w},\mathbf{u}^*) - \nabla_1 f(\mathbf{w},\mathbf{u})\|_2 \leq L_f\|\mathbf{u}^*-\mathbf{u}\|_2,$$
$$\|\nabla_2 f(\mathbf{w},\mathbf{u}^*) - \nabla_2 f(\mathbf{w},\mathbf{u})\|_2 \leq L_f\|\mathbf{u}^*-\mathbf{u}\|_2,$$
$$\|\nabla_{12} g(\mathbf{w},\mathbf{u}^*)\nabla_{22} g(\mathbf{w},\mathbf{u}^*)^{-1}\|_2 \leq \frac{L_g}{\mu_g}.$$

Thus, we have

$$\begin{aligned}&\|\nabla F(\mathbf{w}) - \nabla_1 \bar{F}(\mathbf{w},\mathbf{u}) + \nabla_{12} g(\mathbf{w},\mathbf{u}^*)^\top \nabla_{22} g(\mathbf{w},\mathbf{u}^*)^{-1}\nabla_2 \bar{F}(\mathbf{w},\mathbf{u})\|_2 \\ &\leq L_f(1+\frac{L_g}{\mu_g})\|\mathbf{u}-\mathbf{u}^*\|_2 + L_{gg}\lambda(1+\frac{L_g}{\mu_g})\|\mathbf{u}-\mathbf{u}^*\|_2^2.\end{aligned}$$

Plugging $\mathbf{u} = \mathbf{u}_\lambda^*(\mathbf{w}) = \min_{\mathbf{u}} \bar{F}(\mathbf{w},\mathbf{u})$, then $\nabla_2 \bar{F}(\mathbf{w},\mathbf{u}_\lambda^*(\mathbf{w})) = 0$ and then we have

$$\begin{aligned}&\|\nabla F(\mathbf{w}) - \nabla_1 \bar{F}(\mathbf{w},\mathbf{u}_\lambda^*(\mathbf{w}))\|_2 \\ &\leq L_f(1+\frac{L_g}{\mu_g})\|\mathbf{u}_\lambda^*(\mathbf{w})-\mathbf{u}^*\|_2 + L_{gg}\lambda(1+\frac{L_g}{\mu_g})\|\mathbf{u}_\lambda^*(\mathbf{w})-\mathbf{u}^*\|_2^2.\end{aligned}$$

□

Next, we bound $\|\mathbf{u}_\lambda^*(\mathbf{w}) - \mathbf{u}^*(\mathbf{w})\|_2$.

Lemma 4.27 *Under Assumption 4.10(ii), we have* $\|\mathbf{u}_\lambda^*(\mathbf{w}) - \mathbf{u}^*(\mathbf{w})\|_2 \leq \frac{G_f}{\lambda\mu_g}$.

Proof. By the definitions of $\mathbf{u}_\lambda^*(\mathbf{w}), \mathbf{u}^*(\mathbf{w})$, we have

$$\mathbf{u}_\lambda^*(\mathbf{w}) = \arg\min_{\mathbf{u}} \frac{1}{\lambda} f(\mathbf{w},\mathbf{u}) + g(\mathbf{w},\mathbf{u})$$
$$\mathbf{u}^*(\mathbf{w}) = \arg\min_{\mathbf{u}} g(\mathbf{w},\mathbf{u}).$$

By the optimality condition,

$$\frac{1}{\lambda}\nabla_2 f(\mathbf{w},\mathbf{u}_\lambda^*(\mathbf{w})) + \nabla_2 g(\mathbf{w},\mathbf{u}_\lambda^*(\mathbf{w})) = 0$$
$$\nabla_2 g(\mathbf{w},\mathbf{u}^*(\mathbf{w})) = 0.$$

Since $g(\mathbf{w},\mathbf{u})$ is μ_g-strongly convex w.r.t $\mathbf{u}$ for any $\mathbf{w}$, then we have

$$
\begin{aligned}
g(\mathbf{w}, \mathbf{u}_\lambda^*(\mathbf{w})) \geq & g(\mathbf{w}, \mathbf{u}^*(\mathbf{w})) + \nabla_2 g(\mathbf{w}, \mathbf{u}^*(\mathbf{w}))^\top (\mathbf{u}_\lambda^*(\mathbf{w}) - \mathbf{u}^*(\mathbf{w})) \\
& + \frac{\mu_g}{2} \|\mathbf{u}_\lambda^*(\mathbf{w}) - \mathbf{u}^*(\mathbf{w})\|_2^2 \\
g(\mathbf{w}, \mathbf{u}^*(\mathbf{w})) \geq & g(\mathbf{w}, \mathbf{u}_\lambda^*(\mathbf{w})) + \nabla_2 g(\mathbf{w}, \mathbf{u}_\lambda^*(\mathbf{w}))^\top (\mathbf{u}^*(\mathbf{w}) - \mathbf{u}_\lambda^*(\mathbf{w})) \\
& + \frac{\mu_g}{2} \|\mathbf{u}_\lambda^*(\mathbf{w}) - \mathbf{u}^*(\mathbf{w})\|_2^2.
\end{aligned}
$$

Adding these two inequalities yields:

$$
\begin{aligned}
\mu_g \|\mathbf{u}_\lambda^*(\mathbf{w}) - \mathbf{u}^*(\mathbf{w})\|_2^2 &\leq -\nabla_2 g(\mathbf{w}, \mathbf{u}_\lambda^*(\mathbf{w}))^\top (\mathbf{u}^*(\mathbf{w})) - \mathbf{u}_\lambda^*(\mathbf{w}))) \\
&= \frac{1}{\lambda} \nabla_2 f(\mathbf{w}, \mathbf{u}_\lambda^*(\mathbf{w}))^\top (\mathbf{u}^*(\mathbf{w})) - \mathbf{u}_\lambda^*(\mathbf{w}))) \\
&\leq \frac{1}{\lambda} \|\nabla_2 f(\mathbf{w}, \mathbf{u}_\lambda^*(\mathbf{w}))\|_2 \|(\mathbf{u}^*(\mathbf{w})) - \mathbf{u}_\lambda^*(\mathbf{w}))\|_2.
\end{aligned}
$$

Dividing both sides by $\|\mathbf{u}^*(\mathbf{w})) - \mathbf{u}_\lambda^*(\mathbf{w})\|_2$ and noting $\|\nabla_2 f(\mathbf{w}, \mathbf{u}_\lambda^*(\mathbf{w}))\|_2 \leq G_f$ concludes the proof.

□

Corollary 4.2 *Under the same setting as in Theorem 4.6 with $\lambda = O(\frac{1}{\epsilon}) > 2L_f/\mu_g$ and assume $\|\mathbf{y}_0 - \mathbf{y}^*(\mathbf{w}_0)\|_2^2 \leq O(\epsilon)$, then the following holds*

$$
\mathbb{E}\left[\|\nabla F(\mathbf{w}_\tau)\|_2\right] \leq O(\epsilon), \tag{4.89}
$$

with an iteration complexity of

$$
T = O\left(\max\left\{\frac{1}{\epsilon^3}, \frac{\sigma_f^2}{\epsilon^5}, \frac{\sigma_g^2}{\epsilon^7}\right\}\right), \tag{4.90}
$$

where $\tau \in \{0, \ldots, T-1\}$ is randomly sampled.

Proof. Combining Lemma 4.25 and Lemma 4.27, we have

$$
\begin{aligned}
\|\nabla F(\mathbf{w}_\tau)\|_2 &= \|\nabla F(\mathbf{w}_\tau) - \nabla_1 \bar{F}(\mathbf{w}_\tau, \mathbf{u}_\lambda^*(\mathbf{w}_\tau))\|_2 + \|\nabla_1 \bar{F}(\mathbf{w}_\tau, \mathbf{u}_\lambda^*(\mathbf{w}_\tau))\|_2 \\
&\leq L_f(1 + \frac{L_g}{\mu_g}) \frac{G_f}{\mu_g \lambda} + L_{gg} \lambda (1 + \frac{L_g}{\mu_g}) \frac{G_f^2}{\mu_g^2 \lambda^2} \\
&+ \|\nabla_1 \bar{F}(\mathbf{w}_\tau, \mathbf{u}_\lambda^*(\mathbf{w}_\tau)) - \nabla_1 \bar{F}(\mathbf{w}_\tau, \mathbf{u}_\tau)\|_2 + \|\nabla_1 \bar{F}(\mathbf{w}_\tau, \mathbf{u}_\tau)\|_2 \\
&\leq \frac{2L_f L_g G_f}{\mu_g^2 \lambda} + \frac{2L_{gg} L_g G_f^2}{\mu_g^3 \lambda} \\
&+ \|\nabla_1 \bar{F}(\mathbf{w}_\tau, \mathbf{u}_\lambda^*(\mathbf{w}_\tau)) - \nabla_1 \bar{F}(\mathbf{w}_\tau, \mathbf{u}_\tau)\|_2 + \|\nabla_1 \bar{F}(\mathbf{w}_\tau, \mathbf{u}_\tau)\|_2.
\end{aligned}
$$

Since $\bar{F}(\mathbf{w}, \mathbf{u})$ is $(\lambda\mu_g - L_f)$-strongly convex w.r.t $\mathbf{u}$, Lemma 1.6(c) implies that

$$(\lambda\mu_g - L_f)\|\mathbf{u}_\lambda^*(\mathbf{w}_\tau) - \mathbf{u}_\tau\|_2^2 \leq \frac{1}{(\lambda\mu_g - L_f)}\|\nabla_2\bar{F}(\mathbf{w}_\tau, \mathbf{u}_\tau) - \nabla_2\bar{F}(\mathbf{w}_\tau, \mathbf{u}_\lambda^*(\mathbf{w}_\tau))\|_2^2$$
$$= \frac{1}{(\lambda\mu_g - L_f)}\|\nabla_2\bar{F}(\mathbf{w}_\tau, \mathbf{u}_\tau)\|_2^2$$

Due to $\nabla_1\bar{F}(\mathbf{w}, \mathbf{u}) = \nabla_1 f(\mathbf{w}, \mathbf{u}) + \lambda(\nabla_1 g(\mathbf{w}, \mathbf{u}) - \nabla_1 g(\mathbf{w}, \mathbf{u}^*(\mathbf{w})))$, we have

$$\|\nabla_1\bar{F}(\mathbf{w}_\tau, \mathbf{u}_\lambda^*(\mathbf{w}_\tau)) - \nabla_1\bar{F}(\mathbf{w}_\tau, \mathbf{u}_\tau)\|_2 \leq (L_f + \lambda L_g)\|\mathbf{u}_\lambda^*(\mathbf{w}_\tau) - \mathbf{u}_\tau\|_2$$
$$\leq \frac{(L_f + \lambda L_g)}{(\lambda\mu_g - L_f)}\|\nabla_2\bar{F}(\mathbf{w}_\tau, \mathbf{u}_\tau)\|_2$$
$$\leq \frac{2(\lambda\mu_g/2 + \lambda L_g)}{\lambda\mu_g}\|\nabla_2\bar{F}(\mathbf{w}_\tau, \mathbf{u}_\tau)\|_2 = \frac{\mu_g + 2L_g}{\mu_g}\|\nabla_2\bar{F}(\mathbf{w}_\tau, \mathbf{u}_\tau)\|_2$$

where the last inequality uses $L_f \leq \lambda\mu_g/2$. Combining the above inequalities, we obtain

$$\|\nabla F(\mathbf{w}_\tau)\|_2 \leq \frac{2L_f L_g G_f}{\mu_g^2\lambda} + \frac{2L_{gg}L_g G_f^2}{\mu_g^3\lambda}$$
$$+ \frac{\mu_g + 2L_g}{\mu_g}\|\nabla_2\bar{F}(\mathbf{w}_\tau, \mathbf{u}_\tau)\|_2 + \|\nabla_1\bar{F}(\mathbf{w}_\tau, \mathbf{u}_\tau)\|_2.$$

From Theorem 4.6, we have

$$\mathbb{E}\big[\|\nabla_2\bar{F}(\mathbf{w}_\tau, \mathbf{u}_\tau)\|_2^2 + \|\nabla_1\bar{F}(\mathbf{w}_\tau, \mathbf{u}_\tau)\|_2^2\big] \leq \epsilon^2.$$

Hence, it follows that $\mathbb{E}[\|\nabla_2\bar{F}(\mathbf{w}_\tau, \mathbf{u}_\tau)\|_2] \leq \epsilon$ and $\mathbb{E}[\|\nabla_1\bar{F}(\mathbf{w}_\tau, \mathbf{u}_\tau)\|_2] \leq \epsilon$. If $\lambda = O(1/\epsilon)$, then $\mathbb{E}[\|\nabla F(\mathbf{w}_\tau)\|_2] \leq O(\epsilon)$. The iteration complexity can be established by substituting $\lambda = O(1/\epsilon)$ into Theorem 4.6 and noting that $C_\Upsilon = O(1)$ when $\|\mathbf{y}_0 - \mathbf{y}^*(\mathbf{w}_0)\|_2^2 \leq O(\epsilon)$.

□

Critical: The complexity of $O(1/\epsilon^7)$ is not the state-of-the-art sample complexity achievable under the same assumptions. Indeed, a double-loop large-batch method—similar to the one presented in Section 4.5.1.1 for solving the min-max problem $\min_{\bar{\mathbf{w}}} \max_{\mathbf{y}} \bar{f}(\bar{\mathbf{w}}, \mathbf{y})$—can yield a superior sample complexity of $O(1/\epsilon^6)$ for achieving the stationarity condition $\mathbb{E}[\|\nabla F(\mathbf{w})\|_2] \leq \epsilon^2$. To see this, we apply the results from Section 4.5.1.1, which indicates that a sample complexity for achieving $\mathbb{E}[\|\nabla \bar{F}(\bar{\mathbf{w}})\|_2^2] \leq \epsilon^2$ is $O\left(\frac{\bar{L}_F \bar{\sigma}_1^2}{\epsilon^4} + \frac{\bar{L}_F \bar{L}_1^2 \bar{\sigma}_2^2}{\bar{\mu}^2 \epsilon^4}\right)$. Here, $\bar{L}_F$ denotes the smoothness constant of the objective function $\bar{F}(\bar{\mathbf{w}}) = \max_{\mathbf{y}} \bar{f}(\bar{\mathbf{w}}, \mathbf{y})$. The remaining parameters are defined as follows:

- $\bar{L}_1 = O(\lambda)$ is the Lipschitz constant of $\nabla_1 \bar{f}(\cdot, \cdot)$;
- $\bar{\mu} = O(\lambda)$ is the strong concavity parameter of $\bar{f}(\cdot, \mathbf{y})$ with respect to $\mathbf{y}$;
- $\bar{\sigma}_2^2 = O(\lambda^2)$ represents the variance of the stochastic gradient with respect to $\mathbf{y}$;
- $\bar{\sigma}_1^2 = O(\lambda^2)$ is the variance of the stochastic gradient with respect to $\bar{\mathbf{w}} = (\mathbf{w}, \mathbf{u})$.

Given that we can establish $\bar{L}_F = O(1)$ independent of λ (Chen et al., 2025a, see Lemma B.7) and $\lambda = O(1/\epsilon)$, the total sample complexity reduces to $O(1/\epsilon^6)$.
However, it remains an open problem to develop a single-loop stochastic algorithm that achieves $O(1/\epsilon^6)$ complexity without requiring a large batch size or assuming mean-square smoothness (see next section for more discussion).

4.6 History and Notes

The optimization techniques presented in this chapter for stochastic compositional optimization are rooted in the pioneering work of Yuri Ermoliev (Ermoliev, 1976; Ermoliev and Wets, 1988). The monograph (Ermoliev, 1976), written in Ukrainian, laid the early foundations. Chapter 6 of the edited volume (Ermoliev and Wets, 1988) introduces an early form of the Stochastic Compositional Gradient Descent (SCGD) method, employing a sequence of moving average estimators $\mathbf{u}_t$ to track the inner function values at each iteration—referred to then simply as "averaging." The convergence analysis in these early works is largely limited to asymptotic results, if provided at all. Notably, these works considered a broader class of problems with functional constraints, which will be discussed further in Chapter 6.

The study of non-smooth compositional optimization, where a non-smooth convex function is composed with a smooth function, was first initiated in the works of Fletcher and Watson (1980); Fletcher (1982). Their proposed method, known as the *prox-linear method*, has since been extensively studied and developed in subsequent research (Lewis and Wright, 2009; Duchi and Ruan, 2018; Drusvyatskiy et al.,

2021; Duchi and Ruan, 2017; Drusvyatskiy and Paquette, 2019). We will consider non-smooth compositional optimization in next chapter.

The modern convergence analysis with non-asymptotic rates for stochastic compositional optimization was pioneered by Wang et al. (2017a). Their initial analysis established an $O(1/\epsilon^8)$ complexity for finding an ϵ-stationary solution to a smooth compositional problem, primarily due to suboptimal choices of learning rates. Subsequent works have aimed to improve this convergence rate (Ghadimi et al., 2020; Wang et al., 2017b; Chen et al., 2021a). The improved complexity of $O(1/\epsilon^5)$ for SCGD is derived by following the parameter settings introduced in Qi et al. (2021c). A further refined complexity of $O(1/\epsilon^4)$, under the assumption that the inner function is smooth, was achieved in Chen et al. (2021a). The use of a moving-average gradient estimator to attain the $O(1/\epsilon^4)$ complexity in stochastic compositional optimization is credited to (Ghadimi et al., 2020).

The modern variance-reduction technique for estimating the gradient of a smooth function originates from (Johnson and Zhang, 2013; Mahdavi and Jin, 2013; Zhang et al., 2013), and was inspired by earlier work (Schmidt et al., 2017) that established linear convergence for finite-sum problems with convex and smooth objectives. This technique is now widely known as SVRG. For the objective function $f(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^n f_i(\mathbf{w})$, the SVRG gradient estimator takes the form $\nabla f_i(\mathbf{w}_t) - \nabla f_i(\bar{\mathbf{w}}) + \nabla f(\bar{\mathbf{w}})$, where $\bar{\mathbf{w}}$ is a reference point whose full gradient $\nabla f(\bar{\mathbf{w}})$ is computed periodically.

For non-convex optimization, the variance reduction technique named SPIDER was initiated by Fang et al. (2018), which proposes a gradient estimator $\mathbf{v}_t = \mathbf{v}_{t-1} + \nabla f_i(\mathbf{w}_t) - \nabla f_i(\mathbf{w}_{t-1})$, with $\mathbf{v}$ being periodically re-initialized using either a full gradient or a large-batch gradient. This approach was earlier proposed under the name SARAH for convex optimization in (Nguyen et al., 2017). The technique later evolved into the STORM estimator (Cutkosky and Orabona, 2019), defined as $\mathbf{v}_t = (1-\beta)\mathbf{v}_{t-1} + \beta\nabla f(\mathbf{w}_t;\xi_t) + (1-\beta)\left[\nabla f(\mathbf{w}_t;\xi_t) - \nabla f(\mathbf{w}_{t-1};\xi_t)\right]$, which eliminates the need for periodic re-initialization.

Huo et al. (2018) applied the SVRG technique for finite-sum compositional optimization where both the inner and outer expectation is an average over a finite set. Hu et al. (2019) and Zhang and Xiao (2019) concurrently applied SARAH/SPIDER to compositional optimization with an expectation form and a finite-sum structure, and derived a complexity of $O(1/\epsilon^3)$ for the expectation form and $O(\sqrt{n}/\epsilon^2)$ for a finite-sum structure with n components. Qi et al. (2021a) applied the STORM estimator for SCO with a complexity of $O(1/\epsilon^3)$ and Chen et al. (2021b) applied the STORM estimator to only the inner function estimation for SCO with a complexity of $O(1/\epsilon^4)$.

The capped ℓ_1 norm for sparse regularization was introduced by Zhang (2013). The minimax concave penalty (MCP) was proposed by Zhang (2010), while the smoothly clipped absolute deviation (SCAD) regularizer was introduced by Fan and Li (2001). The proximal mappings for these non-convex regularizers were studied in (Gong et al., 2013). The non-convex piecewise affine regularization method for quantization was proposed by Ma and Xiao (2025). The theoretical analysis presented in Section 4.4 on non-convex optimization with non-convex regularizers fol-

lows the framework established by Xu et al. (2019a), whose results were applied by Deleu and Bengio (2021) to train sparse deep neural networks.

Stochastic weakly-convex–concave min–max optimization with a complexity of $O(1/\epsilon^6)$ was first studied by Rafique et al. (2018). When the problem is weakly-convex and strongly-concave, the complexity can be improved to $O(1/\epsilon^4)$ using double-loop algorithms (Rafique et al., 2018; Yan et al., 2020a). The analysis of SGDA for smooth non-convex min-max optimization was first established by Lin et al. (2020), who derived a complexity of $O(1/\epsilon^4)$ when using a large batch size on the order of $O(1/\epsilon^2)$ for problems that are strongly concave in the dual variable. Without employing a large batch size, the complexity degrades to $O(1/\epsilon^8)$, which also applies to problems lacking strong concavity. The analysis of the single-loop SMDA algorithm was provided by (Guo et al., 2021b), which also established the convergence guarantees for stochastic bilevel optimization using the first approach introduced in Section 4.5.3. A similar convergence result was achieved in Qiu et al. (2020), which employed moving-average gradient estimators for both the primal and dual variables. Chen et al. (2021a) obtained a complexity of $O(1/\epsilon^4)$ for smooth non-convex strongly-concave problems without relying on moving-average gradient estimators, under the stronger assumption that the Hessian/Jacobian matrix is Lipschitz continuous. An improved rate of $O(1/\epsilon^3)$ for smooth non-convex strongly-concave problems was established by (Huang et al., 2022) through the use of STORM estimators.

Bilevel optimization has a long and rich history (Bracken and McGill, 1973). The first complexity analysis of bilevel optimization was initiated by Ghadimi and Wang (2018), who employed the Neumann series to approximate the inverse of the Hessian. Their proposed double-loop stochastic algorithm achieves a sample complexity of $O(1/\epsilon^6)$ for solving the lower-level problem and $O(1/\epsilon^4)$ for the upper-level problem. Subsequent research has led to improved complexity bounds: $O(1/\epsilon^5)$ in (Hong et al., 2020), $O(1/\epsilon^4)$ in (Ji et al., 2020; Guo et al., 2021b; Chen et al., 2021a), and further down to $O(1/\epsilon^3)$ in (Yang et al., 2021; Khanduri et al., 2021; Guo et al., 2021a) under mean-square smoothness conditions. The analysis corresponding to Approach 1 in Section 4.5.3 can be found in (Qiu et al., 2022), while that of Approach 2 is provided in (Guo et al., 2021b).

Penalty-based metholds for bilevel optimization date back to (Ye et al., 1997), with recent developments appearing in (Liu et al., 2021, 2022; Shen and Chen, 2023). Lemma 4.26 is due to Kwon et al. (2023), which established a sample complexity of $O(1/\epsilon^7)$—comparable to Theorem 4.6—for a different double-loop algorithm. They also derived a complexity of $O(1/\epsilon^6)$ for an algorithm similar to update (4.77), except that the gradient estimators for both the lower- and upper-level functions are replaced with STORM estimators under stronger mean-square smoothness assumptions.

The complexity of $O(1/\epsilon^4)$ for stochastic compositional optimization is known to be optimal, as it matches the lower bound established for standard stochastic optimization (Arjevani et al., 2022). Moreover, under mean-square smoothness assumptions, a reduced complexity of $O(1/\epsilon^3)$ is also proven to be optimal (Arjevani et al., 2022).

Chapter 5
Advances: Finite-sum Coupled Compositional Optimization

Abstract In this chapter, we study a novel family of stochastic compositional optimization problems namely **finite-sum coupled compositional optimization (FCCO)**, and introduce algorithms for solving them. These algorithms have direct applications in addressing the empirical X-risk minimization challenges discussed in Chapter 2. To ensure broad applicability, we examine various settings of this problem, characterized by different properties of outer and inner functions, including smooth and non-smooth cases, as well as convex, weakly convex, and non-convex scenarios. The results presented here also significantly extend and complement those discussed in Chapter 4. We also discuss how to efficiently optimize compositional optimized certainty equivalent risks, especially compositional entropic risk.

Coupling reveals depth where composition meets reality!

Contents

5.1 Finite-sum Coupled Compositional Optimization

Specifically, we focus on the following optimization problem:

$$\min_{\mathbf{w}\in\mathbb{R}^d} F(\mathbf{w}) := \frac{1}{n}\sum_{i=1}^{n} f_i\big(\mathbb{E}_{\zeta\sim\mathbb{P}_i} g_i(\mathbf{w};\zeta)\big), \tag{5.1}$$

where $g_i(\cdot;\zeta) : \mathbb{R}^d \to \mathbb{R}^{d'}$ is a stochastic mapping, $f_i(\cdot) : \mathbb{R}^{d'} \to \mathbb{R}$ is a deterministic function, and $\mathbb{P}_i$ denotes the distribution of the random variable ζ.

We refer to this problem as **finite-sum coupled compositional optimization (FCCO)**. If we interpret i as an outer random variable, a distinctive feature that sets FCCO apart from standard stochastic compositional optimization (SCO) is that each inner stochastic function $g_i(\mathbf{w};\zeta)$ depends on both an inner random variable ζ and an outer index i, giving rise to the term *coupled*. While this problem can be cast as a special case of SCO by defining $f(\mathbf{g}) = \frac{1}{n}\sum_{i=1}^{n} f_i(g_i)$ and $\mathbf{g}(\mathbf{w}) = [g_1(\mathbf{w}), \ldots, g_n(\mathbf{w})]$, the high dimensionality of $\mathbf{g}$ due to large n, along with its stochastic components, significantly complicates the construction of unbiased estimators and theoretical analysis. Therefore, FCCO warrants the development of specialized optimization methods.

Below, we revisit several applications of FCCO in ML and discuss the properties of f_i and g_i.

Group DRO

In Section 2.2.3, we have formulated the CVaR divergence regularized group DRO as

$$\min_{\mathbf{w},\nu} \frac{1}{K\alpha}\sum_{i=1}^{K}[L_i(\mathbf{w}) - \nu]_+ + \nu, \tag{5.2}$$

where $\alpha \in (0,1)$, $L_i(\mathbf{w}) = \frac{1}{n_k}\sum_{j=1}^{n_k} \ell(\mathbf{w};\mathbf{x}_j^i, y_j^i)$ denotes the average loss over data from the i-th group. The first term above is an instance of the FCCO objective, where the outer function $f(g) = ([g]_1 - [g]_2)_+$ is a **convex but non-smooth** function of g, and each inner function $g_i(\mathbf{w},\nu) = [L_i(\mathbf{w}), \nu]^\top$ could be convex or non-convex, smooth or non-smooth depending on applications.

AP Maximization

In Section 2.3.2, the AP maximization has been formulated as the following problem:

$$\min_{\mathbf{w}} \frac{1}{n_+}\sum_{\mathbf{x}_i\in\mathcal{S}_+} f(g_i(\mathbf{w})), \tag{5.3}$$

where $\mathcal{S}_+$ is the set of n_+ positive examples, $g_i(\mathbf{w}) = [g_1(\mathbf{w}; \mathbf{x}_i, \mathcal{S}), g_2(\mathbf{w}; \mathbf{x}_i, \mathcal{S})]^\top$ is a vector mapping with two components:

$$g_1(\mathbf{w}; \mathbf{x}_i, \mathcal{S}) = \frac{1}{|\mathcal{S}|} \sum_{\mathbf{x}_j \in \mathcal{S}} \mathbb{I}(y_j = 1) \ell(h(\mathbf{w}; \mathbf{x}_j) - h(\mathbf{w}; \mathbf{x}_i))$$
$$g_2(\mathbf{w}; \mathbf{x}_i, \mathcal{S}) = \frac{1}{|\mathcal{S}|} \sum_{\mathbf{x}_j \in \mathcal{S}} \ell(h(\mathbf{w}; \mathbf{x}_j) - h(\mathbf{w}; \mathbf{x}_i)),$$

and $f(\mathbf{g}) = -\frac{[\mathbf{g}]_1}{[\mathbf{g}]_2}$ is simple function. We can see that f is **non-convex and smooth** if the loss value is upper bounded and $\ell(0)$ is lower bounded. The inner mapping $g_i(\mathbf{w})$ could be convex (e.g., a linear model) or non-convex (e.g., a deep model), smooth or non-smooth depending on applications.

Contrastive Representation Learning

The contrastive objective of self-supervised representation learning presented in (2.50), is the following:

$$\min_{\mathbf{w}} \frac{1}{n} \sum_{i=1}^{n} \tau \log \left(\varepsilon + \frac{1}{|\mathcal{S}_i^-|} \sum_{\mathbf{y} \in \mathcal{S}_i^-} \exp((s(\mathbf{w}; \mathbf{x}_i, \mathbf{y}) - s(\mathbf{w}; \mathbf{x}_i, \mathbf{x}_i^+))/\tau) \right).$$

The outer function $f(g) = \tau \log(\varepsilon + g)$ is a non-convex function and smooth when ε is lower bounded. Each inner function g_i is a non-convex function of $\mathbf{w}$ in general.

5.2 Smooth Functions

In this section, we consider a non-convex but smooth objective function $F(\mathbf{w})$ with smooth outer functions. In addition, we assume the inner stochastic functions satisfy the following conditions throughout this section.

Assumption 5.1. *We assume that*

(i) $\mathbb{E}_{\zeta \sim \mathbb{P}_i}[\|g_i(\mathbf{w}; \zeta) - g_i(\mathbf{w})\|_2^2] \leq \sigma_0^2$.
(ii) $\mathbb{E}_{\zeta \sim \mathbb{P}_i}[\|\nabla g_i(\mathbf{w}; \zeta) - \nabla g_i(\mathbf{w})\|_2^2] \leq \sigma_2^2$.
(iii) $\mathbb{E}_{\zeta \sim \mathbb{P}_i}[\|\nabla g_i(\mathbf{w}; \zeta)\|_2^2] \leq G_2^2$.

5.2.1 The SOX Algorithm

The first algorithm for solving FCCO is called SOX, named by **S**tochastic **O**ptimization of **X**-risks. Owing to its ease of implementation and favorable practical performance, this algorithm is commonly adopted for addressing FCCO. Below, we outline the assumptions necessary for its analysis.

Assumption 5.2. *There exist* $G_1, L_1, L_F > 0$ *such that*

(i) $f_i : \mathbb{R}^{d'} \mapsto \mathbb{R}$ *is* G_1*-Lipschitz continuous and* L_1*-smooth;*
(ii) $F : \mathbb{R}^d \mapsto \mathbb{R}$ *is* L_F*-smooth;*
(iii) $F_* = \min_{\mathbf{w}} F(\mathbf{w}) \geq > -\infty$.

Similar to that for SCO, we also need to track and estimate the inner functions. However, the difference is that we need to maintain and update n estimators for the n inner functions $g_i(\mathbf{w}), i \in [n]$.

To this end, we maintain n sequence of estimators $\{\mathbf{u}_{i,t}, t \in [T]\}_{i=1}^n$. At the t-th iteration, we draw a set of B random indices $\mathcal{B}_t \subset [n]$ with $|\mathcal{B}_t| = B$. We update $\mathbf{u}_{i,t}, i \in [n]$ by the following:

$$\mathbf{u}_{i,t} = \begin{cases} (1-\gamma_t)\mathbf{u}_{i,t-1} + \gamma_t g_i(\mathbf{w}_t; \zeta_{i,t}), & i \in \mathcal{B}_t \\ \mathbf{u}_{i,t-1}, & \text{o.w.} \end{cases}, t = 1, \ldots, T, \tag{5.4}$$

where $\zeta_{i,t} \sim \mathbb{P}_i$ is a random variable. We refer to the above estimator as coordinate moving average estimator. Then, similar to SCMA, a moving average estimator of the gradient is computed by:

$$\begin{aligned} &\mathbf{v}_t = (1-\beta_t)\mathbf{v}_{t-1} + \beta_t \mathbf{z}_t, \\ &\text{where} \quad \mathbf{z}_t = \frac{1}{|\mathcal{B}_t|} \sum_{i \in \mathcal{B}_t} \nabla g_i(\mathbf{w}_t; \zeta'_{i,t}) \nabla f_i(\mathbf{u}_{i,t}). \end{aligned}$$

Then, the model parameters are updated by:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{v}_t.$$

The detailed steps are presented in Algorithm 14.

Convergence Analysis

Let us first define two notations:

$$\Delta_t = \|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|_2^2, \tag{5.5}$$

$$\delta_t = \frac{1}{n} \sum_{i=1}^n \left\|\mathbf{u}_{i,t} - g_i(\mathbf{w}_t)\right\|_2^2. \tag{5.6}$$

Algorithm 14 SOX

Input: learning rate schedules $\{\eta_t\}_{t=1}^T$, $\{\gamma_t\}_{t=1}^T$, $\{\beta_t\}_{t=1}^T$; starting points $\mathbf{w}_0$, $\mathbf{u}_0$, $\mathbf{v}_0$
Let $\mathbf{w}_1 = \mathbf{w}_0 - \eta_0 \mathbf{v}_0$
for $t = 1, \ldots, T$ **do**
  Draw a batch of samples $\mathcal{B}_t \subset [n]$
  for $i \in \mathcal{B}_t$ **do**
    Draw two samples $\zeta_{i,t}, \zeta'_{i,t} \sim \mathbb{P}_i$
    Update the inner function value estimators

$$\mathbf{u}_{i,t} = (1 - \gamma_t)\mathbf{u}_{i,t-1} + \gamma_t g_i(\mathbf{w}_t; \zeta_{i,t}),$$

  end for
  Set $\mathbf{u}_{i,t} = \mathbf{u}_{i,t-1}, i \notin \mathcal{B}_t$
  Compute the vanilla gradient estimator $\mathbf{z}_t = \frac{1}{|\mathcal{B}_t|} \sum_{i \in \mathcal{B}_t} \nabla g_i(\mathbf{w}_t; \zeta'_{i,t}) \nabla f_i(\mathbf{u}_{i,t})$
  Update the MA gradient estimator $\mathbf{v}_t = (1 - \beta_t)\mathbf{v}_{t-1} + \beta_t \mathbf{z}_t$
  Update the model $\mathbf{w}$ by $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{v}_t$
end for

The descent lemma (Lemma 4.9) remains valid. Next, we analyze the recursion of Δ_t and δ_t. One point of deviation is that only some randomly selected coordinates of $\mathbf{u}$ are updated and used for computing the gradient estimator $\mathbf{z}_t$. To facilitate the proof, we introduce a virtual sequence:

$$\bar{\mathbf{u}}_{i,t} = (1 - \gamma_t)\mathbf{u}_{i,t-1} + \gamma_t g_i(\mathbf{w}_t; \zeta_{i,t}), \forall i = 1, \ldots, n. \tag{5.7}$$

This is similar to that is done in the analysis of stochastic coordinate descent method in Section 3.3. Then, we have

$$\mathcal{M}_t = \mathbb{E}_{\mathcal{B}_t, \zeta'_t}[\mathbf{z}_t] = \frac{1}{n} \sum_{i=1}^n \nabla g_i(\mathbf{w}_t) \nabla f_i(\bar{\mathbf{u}}_{i,t}).$$

Critical: Since $\mathbf{u}_t$ is a random variable that depends on $\mathcal{B}_t$, hence

$$\mathbb{E}_{\mathcal{B}_t, \zeta'_t}[\mathbf{z}_t] \neq \frac{1}{n} \sum_{i=1}^n \nabla g_i(\mathbf{w}_t) \nabla f_i(\mathbf{u}_{i,t}).$$

We first bound the error recursion of δ_t.

Lemma 5.1 *Consider the* $\mathbf{u}_t$ *updates in Algorithm 14. Under Assumption 5.1, if* $\gamma_t \leq 1$*, then*

$$\mathbb{E}[\delta_t] \leq \left(1 - \frac{B\gamma_t}{2n}\right) \mathbb{E}[\delta_{t-1}] + \frac{2nG_2^2}{B\gamma_t} \mathbb{E}\left[\|\mathbf{w}_{t-1} - \mathbf{w}_t\|_2^2\right] + \frac{B\gamma_t^2 \sigma_0^2}{n}.$$

Proof. Since $\bar{\mathbf{u}}_{i,t}$ is updated using MA, then similar to (4.6), for all $i \in [n]$ we have

$$\mathbb{E}_{\zeta_{i,t}}[\|\bar{\mathbf{u}}_{i,t} - g_i(\mathbf{w}_t)\|_2^2] \le (1-\gamma_t)^2\|\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_t)\|_2^2 + \gamma_t^2\sigma_0^2.$$

Given $i \in [n]$, with a probability of B/n that $i \in \mathcal{B}_t$, we have $\mathbf{u}_{i,t} = \bar{\mathbf{u}}_{i,t}$; otherwise, $\mathbf{u}_{i,t} = \mathbf{u}_{i,t-1}$. Hence,

$$\begin{aligned}
&\mathbb{E}_{\zeta_{i,t}}\mathbb{E}_{\mathcal{B}_t}[\|\mathbf{u}_{i,t} - g_i(\mathbf{w}_t)\|_2^2] \\
&= \frac{B}{n}\mathbb{E}_{\zeta_{i,t}}[\|\bar{\mathbf{u}}_{i,t} - g_i(\mathbf{w}_t)\|_2^2] + (1-\frac{B}{n})\|\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_t)\|_2^2 \\
&\le \frac{B}{n}(1-\gamma_t)^2\|\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_t)\|_2^2 + \frac{B\gamma_t^2\sigma_0^2}{n} + (1-\frac{B}{n})\|\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_t)\|_2^2 \\
&\le (1-\frac{B\gamma_t}{2n})^2\|\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_t)\|_2^2 + \frac{B\gamma_t^2\sigma_0^2}{n},
\end{aligned}$$

where we use the fact $\frac{B}{n}(1-\gamma_t)^2 + (1-\frac{B}{n}) \le (1-\frac{\gamma_t B}{2n})^2$. Then, taking expectation over all randomness on both sides yields

$$\mathbb{E}\left[\|\mathbf{u}_{i,t} - g_i(\mathbf{w}_t)\|_2^2\right] \le (1-\frac{B\gamma_t}{2n})^2\mathbb{E}\left[\|\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_t)\|_2^2\right] + \frac{B\gamma_t^2\sigma_0^2}{n}.$$

Then using the Young's inequality similar to the proof of Lemma 4.1, we have

$$\begin{aligned}
\mathbb{E}\left[\|\mathbf{u}_{i,t} - g_i(\mathbf{w}_t)\|_2^2\right] &\le (1+\frac{B\gamma_t}{2n})(1-\frac{B\gamma_t}{2n})^2\mathbb{E}\left[\|\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_{t-1})\|_2^2\right] \\
&+ (1+\frac{2n}{B\gamma_t})(1-\frac{B\gamma_t}{2n})^2\mathbb{E}\left[\|g_i(\mathbf{w}_{t-1}) - g_i(\mathbf{w}_t)\|_2^2\right] + \frac{B\gamma_t^2\sigma_0^2}{n} \\
&\le (1-\frac{B\gamma_t}{2n})\mathbb{E}\left[\|\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_{t-1})\|_2^2\right] + \frac{2nG_2^2}{B\gamma_t}\mathbb{E}\left[\|\mathbf{w}_{t-1} - \mathbf{w}_t\|_2^2\right] + \frac{B\gamma_t^2\sigma_0^2}{n},
\end{aligned}$$

where we use $\gamma_t \le 1 < \frac{2n}{B}$. The desired result follows by taking average over $i = 1, \ldots, n$ on both sides.

□

Lemma 5.2 (Variance of $\mathbf{z}_t$) *Let* $\sigma^2 = \frac{G_1^2\sigma_2^2}{B} + \frac{G_1^2G_2^2}{B}\frac{n-B}{n-1}$. *We have*

$$\mathbb{E}_t\left[\|\mathbf{z}_t - \mathcal{M}_t\|_2^2\right] \le \sigma^2.$$

Proof. First, using the variance bound of the average of B independent zero-mean random variables gives

$$A_1 = \mathbb{E}_t\left[\left\|\frac{1}{B}\sum_{i\in\mathcal{B}_t}\nabla g_i(\mathbf{w}_t;\zeta'_{i,t})\nabla f_i(\bar{\mathbf{u}}_{i,t}) - \frac{1}{B}\sum_{i\in\mathcal{B}_t}\nabla g_i(\mathbf{w}_t)\nabla f_i(\bar{\mathbf{u}}_{i,t})\right\|_2^2\right] \le \frac{G_1^2\sigma_2^2}{B},$$

and using the variance bound of B random variables without replacement yields

$$A_2 = \mathbb{E}_t \left[\left\| \frac{1}{B} \sum_{i \in \mathcal{B}_t} \nabla g_i(\mathbf{w}_t) \nabla f_i(\bar{\mathbf{u}}_{i,t}) - \frac{1}{n} \sum_{i=1}^{n} \nabla g_i(\mathbf{w}_t) \nabla f_i(\bar{\mathbf{u}}_{i,t}) \right\|_2^2 \right] \leq \frac{G_1^2 G_2^2}{B} \frac{n-B}{n-1}.$$

As a result,

$$\begin{aligned}
&\mathbb{E}_t \left[\|\mathbf{z}_t - \mathcal{M}_t\|_2^2 \right] \\
&= \mathbb{E}_t \left[\left\| \frac{1}{B} \sum_{i \in \mathcal{B}_t} \nabla g_i(\mathbf{w}_t; \zeta'_{i,t}) \nabla f_i(\bar{\mathbf{u}}_{i,t}) - \frac{1}{n} \sum_{i=1}^{n} \nabla g_i(\mathbf{w}_t) \nabla f_i(\bar{\mathbf{u}}_{i,t}) \right\|_2^2 \right] \\
&= A_1 + A_2 \leq \frac{G_1^2 \sigma_2^2}{B} + \frac{G_1^2 G_2^2}{B} \frac{n-B}{n-1} := \sigma^2.
\end{aligned}$$

□

Lemma 5.3 *Under Assumptions 5.1 and 5.2, if $\beta_t \leq 1$, the gradient estimation error Δ_t can be bounded as*

$$\begin{aligned}
\mathbb{E}[\Delta_t] \leq (1-\beta_t)\mathbb{E}\left[\Delta_{t-1}\right] + \frac{2L_F^2 + 8\beta_t^2 G_2^4 L_1^2}{\beta_t} \mathbb{E}\left[\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2\right] + 8\beta_t L_1^2 G_2^2 \mathbb{E}\left[\delta_{t-1}\right] \\
+ \beta_t^2 \sigma^2 + 4G_2^2 L_1^2 \beta_t \gamma_t^2 \sigma_0^2,
\end{aligned}$$

where $\sigma^2 = \frac{G_1^2 \sigma_2^2}{B} + \frac{G_1^2 G_2^2}{B} \frac{n-B}{n-1}$.

Proof. Since $\mathbf{v}_t$ is updated using MA, we apply Lemma 4.7 in light of Lemma 5.2, yielding

$$\begin{aligned}
\mathbb{E}\left[\|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|_2^2\right] \leq (1-\beta_t)\mathbb{E}[\|\mathbf{v}_{t-1} - \nabla F(\mathbf{w}_{t-1})\|_2^2] \\
+ \frac{2L_F^2}{\beta_t}\mathbb{E}[\|\mathbf{w}_{t-1} - \mathbf{w}_t\|_2^2] + 4\beta_t \mathbb{E}[\|\mathcal{M}_t - \nabla F(\mathbf{w}_t)\|_2^2] + \beta_t^2 \sigma^2.
\end{aligned} \tag{5.8}$$

Next, we bound $\mathbb{E}[\|\mathcal{M}_t - \nabla F(\mathbf{w}_t)\|_2^2]$.

$$\begin{aligned}
\|\mathcal{M}_t - \nabla F(\mathbf{w}_t)\|_2^2 &= \left\| \frac{1}{n} \sum_{i=1}^{n} \nabla g_i(\mathbf{w}_t) \nabla f(\bar{\mathbf{u}}_{i,t}) - \frac{1}{n} \sum_{i=1}^{n} \nabla g_i(\mathbf{w}_t) \nabla f(g_i(\mathbf{w}_t)) \right\|_2^2 \\
&\leq G_2^2 L_1^2 \frac{1}{n} \sum_{i=1}^{n} \|\bar{\mathbf{u}}_{i,t} - g_i(\mathbf{w}_t)\|_2^2.
\end{aligned}$$

From Lemma 5.1, we have

$$\mathbb{E}_{\zeta_{i,t}}[\|\bar{\mathbf{u}}_{i,t} - g_i(\mathbf{w}_t)\|_2^2] \leq (1-\gamma_t)^2 \|\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_t)\|_2^2 + \gamma_t^2 \sigma_0^2, \forall i.$$

Hence

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\|\bar{\mathbf{u}}_{i,t}-g_i(\mathbf{w}_t)\|_2^2\right] \le (1-\gamma_t)^2\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\|\mathbf{u}_{i,t-1}-g_i(\mathbf{w}_t)\|_2^2\right]+\gamma_t^2\sigma_0^2$$
$$\le (1-\gamma_t)^2\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}(2\|g_i(\mathbf{w}_t)-g_i(\mathbf{w}_{t-1})\|_2^2+2\|\mathbf{u}_{i,t-1}-g_i(\mathbf{w}_{t-1})\|_2^2)\right]+\gamma_t^2\sigma_0^2$$
$$\le 2\mathbb{E}[\delta_{t-1}]+\gamma_t^2\sigma_0^2+\mathbb{E}\left[2G_2^2\|\mathbf{w}_{t-1}-\mathbf{w}_t\|_2^2\right].$$

As a result,

$$\mathbb{E}[\|\mathcal{M}_t-\nabla F(\mathbf{w}_t)\|_2^2] \le 2G_2^2L_1^2\mathbb{E}[\delta_{t-1}]+G_2^2L_1^2\gamma_t^2\sigma_0^2+\mathbb{E}\left[2G_2^4L_1^2\|\mathbf{w}_{t-1}-\mathbf{w}_t\|_2^2\right].$$

Plugging the above results into (5.8) we finish the proof.

□

For combining the descent lemma and the above lemmas, we present a result similar to Lemma 4.10, with differences highlighted in boxes.

Lemma 5.4 *If $\eta_t \le 1/L$, assume that there exist non-negative sequences $A_t, B_t, \Gamma_t, \Delta_t, \delta_t, t \ge 0$ satisfying:*

$$(*)A_{t+1} \le A_t+\eta_t\Delta_t-\eta_tB_t-\eta_t\Gamma_t$$
$$(\sharp)\Delta_{t+1} \le (1-\beta_{t+1})\Delta_t+C_1\beta_{t+1}\boxed{\delta_t}+\frac{C_2\eta_t^2}{\beta_{t+1}}\Gamma_t+\beta_{t+1}^2\sigma^2+\boxed{\beta_{t+1}\gamma_{t+1}^2\sigma'^2},$$
$$(\diamond)\delta_{t+1} \le (1-\gamma_{t+1})\delta_t+\frac{C_3\eta_t^2}{\gamma_{t+1}}\Gamma_t+\gamma_{t+1}^2\sigma'^2.$$

If $\beta = \frac{\epsilon^2}{4\sigma^2}, \gamma = \min(\frac{\epsilon^2}{8C_1\sigma'^2}, \frac{\epsilon}{2\sigma''}), \eta = \min(\frac{1}{L}, \frac{\beta}{\sqrt{4C_2}}, \frac{\gamma}{\sqrt{8C_1C_3}})$, then in order to guarantee

$$\sum_{t=0}^{T-1}\frac{1}{T}(B_t+\frac{1}{2}\Gamma_t) \le \epsilon^2.$$

the iteration complexity is in the order of

$$T = O\left(\max\left\{\frac{C_\Upsilon L_F}{\epsilon^2}, \frac{C_\Upsilon\sqrt{C_1C_3}\sigma''}{\epsilon^3}, \frac{C_\Upsilon\sigma^2\sqrt{C_2}}{\epsilon^4}, \frac{C_\Upsilon\sqrt{C_1C_3}C_1\sigma'^2}{\epsilon^4}\right\}\right).$$

where $C_\Upsilon \le A_0-\min_t A_t+\frac{1}{2\sqrt{C_2}}\Delta_0+\sqrt{\frac{C_1}{2C_3}}\delta_0$.

Proof. Following similar analysis to Lemma 4.10, we have

$$A_{t+1} + \frac{\eta_t}{\beta_{t+1}}\Delta_{t+1} + (C_1\eta_t\frac{1+\gamma_{t+1}}{\gamma_{t+1}} - C_1\eta_t)\delta_{t+1} \le A_t - \eta_t B_t - \eta_t\Gamma_t$$
$$+ \left(\eta_t + \frac{\eta_t}{\beta_{t+1}}(1-\beta_{t+1})\right)\Delta_t + \frac{C_2\eta_t^3}{\beta_{t+1}^2}\Gamma_t + \eta_t(\beta_{t+1}\sigma^2 + \boxed{\gamma_{t+1}^2\sigma''^2}) + \boxed{C_1\eta_t(\delta_t - \delta_{t+1})}$$
$$+ C_1\eta_t\frac{1+\gamma_{t+1}}{\gamma_{t+1}}(1-\gamma_{t+1})\delta_t + \frac{C_3C_1\eta_t^3(1+\gamma_{t+1})}{\gamma_{t+1}^2}\Gamma_t + C_1\eta_t(1+\gamma_{t+1})\gamma_{t+1}\sigma'^2.$$

where the terms in the box highlight the difference due to the slight difference in the recursion of Δ_t. Under similar conditions of $\beta_{t+1}, \gamma_{t+1}, \eta_t$ and similar analysis, we get

$$A_{t+1} + \frac{\eta_t}{\beta_{t+1}}\Delta_{t+1} + \frac{C_1\eta_t}{\gamma_{t+1}}\delta_{t+1} \le A_t + \frac{\eta_{t-1}}{\beta_t}\Delta_t + \frac{C_1\eta_{t-1}}{\gamma_t}\delta_t + \boxed{C_1\eta_t(\delta_t - \delta_{t+1})}$$
$$- \eta_t B_t - \frac{1}{2}\eta_t\Gamma_t + \eta_t(\beta_{t+1}\sigma^2 + \boxed{\gamma_{t+1}^2\sigma''^2}) + 2C_1\eta_t\gamma_{t+1}\sigma'^2.$$

Since $\eta_{t+1} \le \eta_t$, we have

$$A_{t+1} + \frac{\eta_t}{\beta_{t+1}}\Delta_{t+1} + \frac{C_1\eta_t}{\gamma_{t+1}}\delta_{t+1} + \boxed{C_1\eta_{t+1}\delta_{t+1}} \le A_t + \frac{\eta_{t-1}}{\beta_t}\Delta_t + \frac{C_1\eta_{t-1}}{\gamma_t}\delta_t + \boxed{C_1\eta_t\delta_t}$$
$$- \eta_t B_t - \frac{1}{2}\eta_t\Gamma_t + \eta_t(\beta_{t+1}\sigma^2 + \boxed{\gamma_{t+1}^2\sigma''^2}) + 2C_1\eta_t\gamma_{t+1}\sigma'^2.$$

Define $\Upsilon_{t+1} = A_{t+1} + \frac{\eta_t}{\beta_{t+1}}\Delta_{t+1} + \frac{C_1\eta_t}{\gamma_{t+1}}\delta_{t+1} + \boxed{C_1\eta_{t+1}\delta_{t+1}}$, we have

$$\eta_t B_t + \frac{1}{2}\eta_t\Gamma_t \le \Upsilon_t - \Upsilon_{t+1} + \eta_t(\beta_{t+1}\sigma^2 + \boxed{\gamma_{t+1}^2\sigma''^2}) + 2C_1\eta_t\gamma_{t+1}\sigma'^2.$$

Hence

$$\sum_{t=0}^{T-1}(\eta_t B_t + \frac{1}{2}\eta_t\Gamma_t) \le \Upsilon_0 - A_* + \sum_{t=0}^{T-1}\left(\eta_t\beta_{t+1}\sigma^2 + 2C_1\eta_t\gamma_{t+1}\sigma'^2 + \eta_t\gamma_{t+1}^2\sigma''^2\right).$$

Next, let us consider $\eta_t = \eta, \beta_t = \beta, \gamma_t = \gamma$. Then we have

$$\sum_{t=0}^{T-1}\frac{1}{T}(B_t + \frac{1}{2}\Gamma_t) \le \frac{C_\Upsilon}{T} + \left(\beta\sigma^2 + 2\gamma C_1\sigma'^2 + \gamma^2\sigma''^2)\right).$$

Since $\eta_t = \eta, \gamma_t = \gamma, \beta_t = \beta$, in order to ensure the RHS is less than ϵ^2, it suffices to have

$$\beta = \frac{\epsilon^2}{4\sigma^2}, \quad \gamma = \min(\frac{\epsilon^2}{8C_1\sigma'^2}, \frac{\epsilon}{2\sigma''}), \quad T \ge \frac{C_\Upsilon}{4\epsilon^2\eta}.$$

Since

$$\eta = \min(\frac{1}{L}, \frac{\beta}{\sqrt{4C_2}}, \frac{\gamma}{\sqrt{8C_1C_3}}).$$

Thus the order of T becomes

$$\begin{aligned} T &= O\left(\max\left\{\frac{C_\Upsilon L_F}{\epsilon^2}, \frac{C_\Upsilon\sqrt{C_2}}{\epsilon^2\beta}, \frac{C_\Upsilon\sqrt{C_1C_3}}{\gamma\epsilon^2}\right\}\right) \\ &= O\left(\max\left\{\frac{C_\Upsilon L_F}{\epsilon^2}, \frac{C_\Upsilon\sqrt{C_1C_3}\sigma''}{\epsilon^3}, \frac{C_\Upsilon\sigma^2\sqrt{C_2}}{\epsilon^4}, \frac{C_\Upsilon\sqrt{C_1C_3}C_1\sigma'^2}{\epsilon^4}\right\}\right) \end{aligned}$$

where

$$C_\Upsilon = A_0 - A_* + \frac{\eta}{\beta}\Delta_0 + \frac{C_1\eta}{\gamma}\delta_0 + C_1\eta\delta_0 \leq A_0 - A_* + \frac{1}{2\sqrt{C_2}}\Delta_0 + 2\frac{\sqrt{C_1}}{\sqrt{8C_3}}\delta_0.$$

□

Finally, we state the convergence of SOX.

Theorem 5.1 *Under Assumption 5.1 and 5.2, SOX with* $\beta = \frac{\epsilon^2}{4\sigma^2} < \frac{1}{4L_1G_2}, \gamma = \min(\frac{\epsilon^2}{64G_2^2L_1^2{\sigma_0}^2}, \frac{n}{2BG_1L_1\sigma_0}), \eta = \min(\frac{1}{2L_F}, \frac{\beta}{2\sqrt{C_2}}, \frac{B\gamma}{n\sqrt{32C_1C_3}})$*, can find* $\mathbf{w}_\tau$ *with* τ *randomly sampled from* $\{1, \ldots, T\}$ *so that* $\mathbb{E}\left[\|\mathbf{v}_\tau\|_2^2 + \|\nabla F(\mathbf{w}_\tau)\|_2^2\right] \leq \epsilon^2$ *with an iteration complexity of*

$$T = O\left(\max\left\{\frac{C_\Upsilon L_F}{\epsilon^2}, \frac{C_\Upsilon L_1^2\sigma_0}{\epsilon^3}, \frac{C_\Upsilon L_F\sigma^2}{\epsilon^4}, \frac{C_\Upsilon L_1^3 n\sigma_0^2}{\epsilon^4 B}\right\}\right),$$

where $C_1 = 8G_2^2L_1, C_2 = 4L_F^2 + 2, C_3 = 2G_2^2, \sigma^2 = \frac{G_1^2\sigma_2^2}{B} + \frac{G_1^2G_2^2}{B}\frac{n-B}{n-1}$*, and* $C_\Upsilon = O(F(\mathbf{w}_0) - F_* + \frac{1}{L_F}\|\mathbf{v}_0 - \nabla F(\mathbf{w}_0)\|_2^2 + L_1\frac{1}{n}\|\mathbf{u}_0 - g(\mathbf{w}_0)\|_2^2)$.

Why it matters

Theorem 5.1 shows that SOX achieves a complexity dominated by $O\left(\frac{C_\Upsilon L_1^3 n\sigma_0^2}{\epsilon^4 B}\right)$, which is comparable to that of SCMA for finding an ϵ-stationary solution. The key difference is that the complexity of SOX is scaled by a factor of n/B, since it must track and estimate n inner functions.

Proof. Assume that ϵ is sufficiently small such that $8\beta^2G_2^2L_1^2 \leq 1$. We have established the following three inequalities:

$$(*)\ \mathbb{E}\left[F(\mathbf{w}_{t+1})\right] \le \mathbb{E}\left[F(\mathbf{w}_t)\right] + \frac{\eta}{2}\mathbb{E}[\Delta_t] - \frac{\eta}{2}\mathbb{E}\left[\|\nabla F(\mathbf{w}_t)\|_2^2\right] - \frac{\eta}{4}\mathbb{E}\left[\|\mathbf{v}_t\|_2^2\right],$$

$$(\sharp)\ \mathbb{E}[\Delta_{t+1}] \le (1-\beta)\mathbb{E}\left[\Delta_t\right] + \frac{2L_F^2+1}{\beta}\eta^2\mathbb{E}\left[\|\mathbf{v}_t\|_2^2\right] + 8\beta L_1^2 G_2^2\mathbb{E}\left[\delta_t\right] + \beta^2\sigma^2 + 4G_2^2L_1^2\beta\gamma^2\sigma_0^2,$$

$$(\diamond)\ \mathbb{E}\left[\delta_{t+1}\right] \le \left(1 - \frac{B\gamma}{2n}\right)\mathbb{E}\left[\delta_t\right] + \frac{2nG_2^2\eta^2}{B\gamma}\mathbb{E}\left[\|\mathbf{v}_t\|_2^2\right] + \frac{B\gamma^2\sigma_0^2}{n}.$$

Let us define $\bar{\gamma} = \frac{B\gamma}{2n}$, the last inequality becomes

$$(\diamond)\ \mathbb{E}\left[\delta_{t+1}\right] \le (1-\bar{\gamma})\,\mathbb{E}\left[\delta_t\right] + \frac{G_2^2\eta^2}{\bar{\gamma}}\mathbb{E}\left[\|\mathbf{v}_t\|_2^2\right] + \frac{4n\bar{\gamma}^2\sigma_0^2}{B}.$$

Define $A_t = 2(F(\mathbf{w}_t) - F(\mathbf{w}_*))$ and $B_t = \|\nabla F(\mathbf{w}_t)\|_2^2$, $\Gamma_t = \|\mathbf{v}_t\|_2^2/2$, $\Delta_t = \|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\|_2^2$, $\delta_t = \frac{1}{n}\|\mathbf{u}_t - g(\mathbf{w}_t)\|_2^2$, and $\Upsilon_t = A_t + \frac{\eta_{t-1}}{\beta_t}\Delta_t + \frac{C_1\eta_{t-1}}{\bar{\gamma}_t}\delta_t$.

Then the three inequalities satisfy that in Lemma 4.10 with $C_1 = 8G_2^2L_1^2, C_2 = 2(2L_F^2+1), C_3 = 2G_2^2, \sigma^2 = \frac{G_1^2\sigma_2^2}{B} + \frac{G_1^2G_2^2}{B}\frac{n-B}{n-1}, \sigma'^2 = \frac{4n\sigma_0^2}{B}, \sigma''^2 = 4G_2^2L_1^2\sigma_0^2$. Then $\eta, \beta, \bar{\gamma}$ satisfy

$$\beta = \frac{\epsilon^2}{4\sigma^2}, \quad \bar{\gamma} = \min\left(\frac{\epsilon^2}{8C_1\sigma'^2}, \frac{\epsilon}{2\sigma''}\right) = \min\left(\frac{\epsilon^2 B}{128G_2^2L_1^2n{\sigma_0}^2}, \frac{\epsilon}{4G_2L_1\sigma_0}\right),$$
$$\eta = \min\left(\frac{1}{2L_F}, \frac{\beta}{\sqrt{4C_2}}, \frac{\bar{\gamma}}{\sqrt{8C_1C_3}}\right).$$

Thus the order of T becomes

$$T = O\left(\max\left\{\frac{C_\Upsilon L_F}{\epsilon^2}, \frac{C_\Upsilon\sqrt{C_1C_3}\sigma''}{\epsilon^3}, \frac{C_\Upsilon\sigma^2\sqrt{C_2}}{\epsilon^4}, \frac{C_\Upsilon\sqrt{C_1C_3}C_1\sigma'^2}{\epsilon^4}\right\}\right)$$
$$= O\left(\max\left\{\frac{C_\Upsilon L_F}{\epsilon^2}, \frac{C_\Upsilon L_1^2\sigma_0}{\epsilon^3}, \frac{C_\Upsilon L_F\sigma^2}{\epsilon^4}, \frac{C_\Upsilon L_1^3n\sigma_0^2}{\epsilon^4 B}\right\}\right),$$

where

$$C_\Upsilon$$
$$\le 2(F(\mathbf{w}_0) - F(\mathbf{w}_*)) + \frac{1}{2\sqrt{C_2}}\|\mathbf{v}_0 - \nabla F(\mathbf{w}_0)\|_2^2 + \frac{\sqrt{C_1}}{\sqrt{2C_3}}\frac{1}{n}\|\mathbf{u}_0 - g(\mathbf{w}_0)\|_2^2$$
$$= 2(F(\mathbf{w}_0) - F(\mathbf{w}_*)) + O\left(\frac{1}{L_F}\right)\|\mathbf{v}_0 - \nabla F(\mathbf{w}_0)\|_2^2 + O(L_1)\frac{1}{n}\|\mathbf{u}_0 - g(\mathbf{w}_0)\|_2^2.$$

□

5.2.2 Multi-block Single-Probe Variance Reduction

In this subsection, we present a second algorithm for solving FCCO with an improved complexity than that of SOX under a stronger condition on g_i. We replace Assumption 5.2 by the following:

Assumption 5.3. *There exist $G_1, L_1, L_2 > 0$ such that*

(i) $f_i : \mathbb{R}^{d'} \mapsto \mathbb{R}$ *is G_1-Lipschitz continuous and L_1-smooth;*
(ii) $\nabla g_i(\cdot, \zeta) : \mathbb{R}^d \mapsto \mathbb{R}^{d'}$ *is mean-squared Lipschitz continuous, i.e.,*

$$\mathbb{E}_\zeta[\|\nabla g_i(\mathbf{w}, \zeta) - \nabla g_i(\mathbf{w}', \zeta)\|_2^2] \leq L_2^2 \|\mathbf{w} - \mathbf{w}'\|_2^2, \forall \mathbf{w}, \mathbf{w}';$$

(iii) $F_* = \min_{\mathbf{w}} F(\mathbf{w}) \geq > -\infty$.

The idea is to leverage advanced variance reduction for tracking both the inner functions and the gradient. A straightforward approach is to change the update of $\mathbf{u}_{i,t-1}$ by using the STORM estimator and do similarly for the gradient estimator. In particular, one may change the update for $\mathbf{u}_{i,t}$ according to STORM:

$$\mathbf{u}_{i,t} = \begin{cases} (1-\gamma_t)\mathbf{u}_{i,t-1} + \gamma_t g_i\left(\mathbf{w}_t; \zeta_{i,t}\right) + \underbrace{(1-\gamma_t)(g_i\left(\mathbf{w}_t; \zeta_{i,t}\right) - g_i\left(\mathbf{w}_{t-1}; \zeta_{i,t}\right))}_{\text{error correction}} & i \in \mathcal{B}_t \\ \mathbf{u}_{i,t-1} & i \notin \mathcal{B}_t \end{cases}. \tag{5.9}$$

However, this naive approach does not work as the standard error correction term marked above only accounts for the randomness in $g_i(\mathbf{w}_t; \zeta_{i,t})$ but not in the randomness caused by sampling $i \in \mathcal{B}_t$.

In order to tackle this challenge, we introduce the following estimator termed multi-block single-probe variance reduction estimator (MSVR):

$$\mathbf{u}_{i,t} = \begin{cases} (1-\gamma_t)\mathbf{u}_{i,t-1} + \gamma_t g_i\left(\mathbf{w}_t; \zeta_{i,t}\right) + \gamma_t'\left(g_i\left(\mathbf{w}_t; \zeta_{i,t}\right) - g_i\left(\mathbf{w}_{t-1}; \zeta_{i,t}\right)\right) & i \in \mathcal{B}_t \\ \mathbf{u}_{i,t-1} & i \notin \mathcal{B}_t \end{cases}. \tag{5.10}$$

The difference from (5.9) lies at the value of γ_t', which is set as $\frac{n-B}{B(1-\gamma_t)} + (1-\gamma_t)$ with $B = |\mathcal{B}_t|$. The MSVR estimator can track multiple functional mappings $(g_1, g_2, \cdots, g_n)$, simultaneously, while the number of sampled blocks B_1 for probing can be as small as one. It is notable that when $B = n$, i.e., all blocks are probed at each iteration, $\gamma_t' = 1 - \gamma_t$ and MSVR reduces to STORM applied to $\mathbf{g}(\mathbf{w})$. The additional factor in γ_t', i.e., $\alpha_t = \frac{n-B}{B(1-\gamma_t)}$ is to account for the randomness in the sampled blocks and noise in those blocks that are not updated.

With $\mathbf{u}_t$, we compute a vanilla gradient estimator by

$$\mathbf{z}_t = \frac{1}{B}\sum_{i\in\mathcal{B}'_t} \nabla g_i(\mathbf{w}_t;\zeta'_{i,t})\nabla f_i(\mathbf{u}_{i,t}),$$

where $\mathcal{B}'_t \subset [n]$ is a mini-batch of B indices independent of $\mathcal{B}_t$.

Similar to SCST, we apply another STORM estimator to estimate

$$\mathcal{M}_t = \mathbb{E}_{\mathcal{B}'_t,\zeta'_t}[\mathbf{z}_t] = \frac{1}{n}\sum_{i=1}^{n} \nabla g_i(\mathbf{w}_t)\nabla f_i(\mathbf{u}_{i,t}),$$

with an extra vanilla gradient estimator at previous iteration:

$$\tilde{\mathbf{z}}_{t-1} = \frac{1}{B}\sum_{i\in\mathcal{B}'_t} \nabla g_i(\mathbf{w}_{t-1};\zeta'_{i,t})\nabla f_i(\mathbf{u}_{i,t-1}).$$

This is computed by the following sequence:

$$\mathbf{v}_t = (1-\beta_t)\mathbf{v}_{t-1} + \beta_t\mathbf{z}_t + (1-\beta_t)(\mathbf{z}_t - \tilde{\mathbf{z}}_{t-1}). \tag{5.11}$$

Then we use $\mathbf{v}_t$ to update the model parameter. The full steps are presented in Algorithm 15.

Critical: We use an independent batch $\mathcal{B}'_t$ because $\mathbf{z}_t$ depends on $\mathbf{u}_t$, which depends on $\mathcal{B}_t$. If we use the same batch $\mathcal{B}_t$ to compute $\mathbf{z}_t$, then

$$\begin{aligned}\mathcal{M}_t &= \mathbb{E}_{\mathcal{B}_t,\zeta'_t}\left[\frac{1}{B}\sum_{i\in\mathcal{B}_t} \nabla g_i(\mathbf{w}_t;\zeta'_{i,t})\nabla f_i(\mathbf{u}_{i,t})\right] \\ &= \mathbb{E}_{\mathcal{B}_t,\zeta'_t}\left[\frac{1}{B}\sum_{i\in\mathcal{B}_t} \nabla g_i(\mathbf{w}_t;\zeta'_{i,t})\nabla f_i(\bar{\mathbf{u}}_{i,t})\right] = \frac{1}{n}\sum_{i=1}^{n} \nabla g_i(\mathbf{w}_t)\nabla f_i(\bar{\mathbf{u}}_{i,t}).\end{aligned}$$

where $\bar{\mathbf{u}}_t$ independent of $\mathcal{B}_t$ is defined in (5.12). However, we cannot construct an unbiased estimator of $\mathcal{M}_{t-1}$ since $\bar{\mathbf{u}}_{t-1}$ is not available in the algorithm.

An alternative approach is that we use $\mathbf{u}_{t-1}$ and $\mathbf{u}_{t-2}$ to compute $\mathbf{z}_t$ and $\tilde{\mathbf{z}}_{t-1}$, respectively, with $\mathcal{B}_t$, i.e.,

$$\begin{aligned}\mathbf{z}_t &= \frac{1}{B}\sum_{i\in\mathcal{B}_t} \nabla g_i(\mathbf{w}_t;\zeta'_{i,t})\nabla f_i(\mathbf{u}_{i,t-1}) \\ \tilde{\mathbf{z}}_{t-1} &= \frac{1}{B}\sum_{i\in\mathcal{B}_t} \nabla g_i(\mathbf{w}_{t-1};\zeta'_{i,t})\nabla f_i(\mathbf{u}_{i,t-2}),\end{aligned}$$

and compute $\mathbf{v}_t$ by

$$\mathbf{v}_t = (1-\beta_t)\mathbf{v}_t + \beta_t\mathbf{z}_t + (1-\beta_t)(\mathbf{z}_t - \tilde{\mathbf{z}}_{t-1}).$$

Algorithm 15 MSVR

Input: learning rate schedules $\{\eta_t\}_{t=1}^T$, $\{\gamma_t\}_{t=1}^T$, $\{\beta_t\}_{t=1}^T$; starting points $\mathbf{w}_0$, $\mathbf{u}_0$, $\mathbf{v}_0$
Let $\mathbf{w}_1 = \mathbf{w}_0 - \eta_0 \mathbf{v}_0$
for $t = 1, \ldots, T$ **do**
  Draw two batches of samples $\mathcal{B}_t, \mathcal{B}_t' \subset [n]$
  for $i \in \mathcal{B}_t$ **do**
    Draw two samples $\zeta_{i,t}, \zeta_{i,t}' \sim \mathbb{P}_i$
    Update the inner function value estimators

$$\mathbf{u}_{i,t} = (1-\gamma_t)\mathbf{u}_{i,t-1} + \gamma_t g_i\left(\mathbf{w}_t; \zeta_{i,t}\right) + \gamma_t'\left(g_i\left(\mathbf{w}_t; \zeta_{i,t}\right) - g_i\left(\mathbf{w}_{t-1}; \zeta_{i,t}\right)\right)$$

  end for
  Set $\mathbf{u}_{i,t} = \mathbf{u}_{i,t-1}, i \notin \mathcal{B}_t$
  Compute the vanilla gradient estimator $\mathbf{z}_t = \frac{1}{B}\sum_{i\in\mathcal{B}_t'} \nabla g_i(\mathbf{w}_t; \zeta_{i,t}')\nabla f_i(\mathbf{u}_{i,t})$
  Compute the extra vanilla gradient estimator $\tilde{\mathbf{z}}_{t-1} = \frac{1}{B}\sum_{i\in\mathcal{B}_t'} \nabla g_i(\mathbf{w}_{t-1}; \zeta_{i,t}')\nabla f_i(\mathbf{u}_{i,t-1})$
  Update the STORM gradient estimator $\mathbf{v}_t = (1-\beta_t)\mathbf{v}_{t-1} + \beta_t \mathbf{z}_t + (1-\beta_t)(\mathbf{z}_t - \tilde{\mathbf{z}}_{t-1})$
  Update the model $\mathbf{w}$ by $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{v}_t$
end for

The converge analysis can be performed similarly with slight modifications.

Convergence Analysis

We first analyze the error recursion of

$$\delta_t = \frac{1}{n}\|\mathbf{u}_t - \mathbf{g}(\mathbf{w}_t)\|_2^2 := \frac{1}{n}\sum_{i=1}^n \|\mathbf{u}_{i,t} - g_i(\mathbf{w}_t)\|_2^2.$$

Similar to the analysis of SOX, we introduce virtual sequences $\bar{\mathbf{u}}_{i,t}, \forall i$:

$$\bar{\mathbf{u}}_{i,t} = (1-\gamma_t)\mathbf{u}_{i,t-1} + \gamma_t g_i\left(\mathbf{w}_t; \zeta_{i,t}\right) + \gamma_t'\left(g_i\left(\mathbf{w}_t; \zeta_{i,t}\right) - g_i\left(\mathbf{w}_{t-1}; \zeta_{i,t}\right)\right), \forall i. \quad (5.12)$$

Lemma 5.5 *Consider the $\mathbf{u}_t$ updates in Algorithm 15. Under Assumption 5.1 and 5.3 (ii), by setting $\gamma_t' = \frac{n-B}{B(1-\gamma_t)} + (1-\gamma_t)$, for $\gamma_t \le \frac{1}{2}$, we have:*

$$\mathbb{E}\left[\delta_t\right] \le \left(1 - \frac{B\gamma_t}{n}\right)\mathbb{E}\left[\delta_{t-1}\right] + \frac{2B}{n}\gamma_t^2\sigma_0^2 + \frac{12nG_2^2}{B}\mathbb{E}\left[\|\mathbf{w}_t - \mathbf{w}_{t-1}\|^2\right].$$

Proof. Let us consider a fixed $i \in [n]$. With a probability B/n that $i \in \mathcal{B}_t$, we have $\mathbf{u}_{i,t} = \bar{\mathbf{u}}_{i,t}$; otherwise $\mathbf{u}_{i,t} = \mathbf{u}_{i,t-1}$. Hence,

$$\mathbb{E}\left[\|\mathbf{u}_{i,t} - g_i(\mathbf{w}_t)\|_2^2\right] = \frac{B}{n}\underbrace{\mathbb{E}\left[\|\bar{\mathbf{u}}_{i,t} - g_i(\mathbf{w}_t)\|_2^2\right]}_{A_1} + (1-\frac{B}{n})\underbrace{\mathbb{E}\left[\|\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_t)\|_2^2\right]}_{A_2}.$$

Note that the first term A_1 in the R.H.B. can be bounded similarly as in Lemma 4.12 for using the STORM estimator by building a recursion with $\|\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_{t-1})\|_2^2$. However, there exists the second term due to the randomness of $\mathcal{B}_t$, which can be decomposed as

$$\begin{aligned}
A_2 &= \mathbb{E}[\|\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_{t-1}) + g_i(\mathbf{w}_{t-1}) - g_i(\mathbf{w}_t)\|_2^2] \\
&= \underbrace{\mathbb{E}[\|\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_{t-1})\|_2^2]}_{A_{21}} + \underbrace{\mathbb{E}[\|g_i(\mathbf{w}_{t-1}) - g_i(\mathbf{w}_t)\|_2^2]}_{A_{22}} \\
&+ \underbrace{\mathbb{E}[2(\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_{t-1}))^\top (g_i(\mathbf{w}_{t-1}) - g_i(\mathbf{w}_t))]}_{A_{23}}.
\end{aligned}$$

The first two terms in RHS (A_{21} and A_{22}) can be easily handled. The difficulty comes from the third term, which cannot be simply bounded by using Young's inequality. If doing so, it will end up with a non-diminishing error of $\mathbf{u}_{i,t}$. To combat this difficulty, we use the additional factor brought by $\gamma_t'(g_i(\mathbf{w}_t; \xi_t^i) - g_i(\mathbf{w}_{t-1}; \xi_t^i))$ in A_1 to cancel A_{23}. This is more clear by the following decomposition of A_1.

$$\begin{aligned}
A_1 = \mathbb{E}[\| &\underbrace{(1-\gamma_t)(\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_{t-1}))}_{A_{11}} + \underbrace{\alpha_t(g_i(\mathbf{w}_t) - g_i(\mathbf{w}_{t-1}))}_{A_{12}} \\
&+ \underbrace{\gamma_t(g_i(\mathbf{w}_t; \zeta_{i,t}) - g_i(\mathbf{w}_t))}_{A_{13}} \\
&+ \underbrace{\gamma_t'(g_i(\mathbf{w}_t; \zeta_{i,t}) - g_i(\mathbf{w}_{t-1}; \xi_{i,t}) - g_i(\mathbf{w}_t) + g_i(\mathbf{w}_{t-1}))}_{A_{14}} \|_2^2],
\end{aligned}$$

where $\alpha_t = \gamma_t' + \gamma_t - 1$. Since $\mathbb{E}_t[A_{13}] = 0, \mathbb{E}_t[A_{14}] = 0$, then we have

$$A_1 \leq \mathbb{E}[\|A_{11} + A_{12}\|_2^2] + \mathbb{E}\left[\|A_{13} + A_{14}\|_2^2\right].$$

In light of the above decomposition, we can bound $\mathbb{E}[\|A_{11} + A_{12}\|_2^2] \leq \mathbb{E}[\|A_{11}\|_2^2 + \|A_{12}\|_2^2 + 2A_{11}^\top A_{12}]$ and $\mathbb{E}[\|A_{13} + A_{14}\|_2^2] \leq 2\mathbb{E}[\|A_{13}\|_2^2] + 2\mathbb{E}[\|A_{14}\|_2^2]$. The resulting term $\mathbb{E}[2A_{11}^\top A_{12}]$ has a negative sign as A_{23}. Hence, by carefully choosing γ_t', we can cancel both terms. Specifically, we have

$$
\begin{aligned}
\frac{B}{n}A_1 &\leq \frac{B}{n}\left(\mathbb{E}[\|A_{11}\|_2^2 + \|A_{12}\|_2^2 + 2A_{11}^\top A_{12}] + 2\mathbb{E}[\|A_{13}\|_2^2] + 2\mathbb{E}[\|A_{14}\|_2^2]\right) \\
&= \mathbb{E}\left[\frac{B}{n}(1-\gamma_t)^2\|\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_{t-1})\|_2^2\right] + \mathbb{E}\left[\frac{B}{n}\alpha_t^2\,\|g_i(\mathbf{w}_t) - g_i(\mathbf{w}_{t-1})\|_2^2\right] \\
&+ \mathbb{E}\left[\frac{B}{n}2\alpha_t(1-\gamma_t)(g_i(\mathbf{w}_t) - g_i(\mathbf{w}_{t-1}))^\top(\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_{t-1}))\right] \\
&+ \mathbb{E}\left[\frac{B}{n}2\gamma_t^2\left\|g_i(\mathbf{w}_t;\zeta_{i,t}) - g_i(\mathbf{w}_t)\right\|_2^2\right] \\
&+ \mathbb{E}\left[\frac{B}{n}2\gamma_t'^2\left\|(g_i(\mathbf{w}_t;\zeta_{i,t}) - g_i(\mathbf{w}_{t-1};\zeta_{i,t}) - g_i(\mathbf{w}_t) + g_i(\mathbf{w}_{t-1}))\right\|_2^2\right].
\end{aligned}
$$

Combining the upper bounds of A_1 and A_2, we have

$$
\begin{aligned}
&\frac{B}{n}A_1 + \frac{n-B}{n}A_2 \\
&\leq \mathbb{E}[\frac{B}{n}(1-\gamma_t)^2\|\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_{t-1})\|_2^2 + \frac{B}{n}\alpha_t^2\,\|g_i(\mathbf{w}_t) - g_i(\mathbf{w}_{t-1})\|_2^2] \\
&+ \mathbb{E}\left[\frac{B}{n}2\alpha_t(1-\gamma_t)(g_i(\mathbf{w}_t) - g_i(\mathbf{w}_{t-1}))^\top(\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_{t-1}))\right] \\
&+ \mathbb{E}\left[\frac{B}{n}2\gamma_t^2\left\|g_i(\mathbf{w}_t;\zeta_{i,t}) - g_i(\mathbf{w}_t)\right\|_2^2\right] \\
&+ \mathbb{E}\left[\frac{B}{n}2(\gamma_t')^2\left\|(g_i(\mathbf{w}_t;\zeta_{i,t}) - g_i(\mathbf{w}_{t-1};\zeta_{i,t}) - g_i(\mathbf{w}_t) + g_i(\mathbf{w}_{t-1}))\right\|_2^2\right] \\
&+ \mathbb{E}\left[\frac{n-B}{n}\|\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_{t-1})\|_2^2] + \mathbb{E}[\frac{n-B}{n}\|g_i(\mathbf{w}_{t-1}) - g_i(\mathbf{w}_t)\|_2^2\right] \\
&+ \mathbb{E}\left[\frac{n-B}{n}2(\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_{t-1}))^\top(g_i(\mathbf{w}_{t-1}) - g_i(\mathbf{w}_t))\right].
\end{aligned}
$$

Since $\frac{B}{n}2\alpha_t(1-\gamma_t) = 2\frac{B}{n}\frac{(n-B)}{B(1-\gamma_t)}(1-\gamma_t) = 2\frac{n-B}{n}$, then cross terms will cancel out. The remaining terms can be merged and handled separately. First,

$$
\begin{aligned}
&\mathbb{E}\left[\frac{B}{n}(1-\gamma_t)^2\|\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_{t-1})\|_2^2 + \frac{n-B}{n}\|\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_{t-1})\|_2^2\right] \\
&\leq (1 - \frac{B}{n}\gamma_t)\mathbb{E}[\|\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_{t-1})\|_2^2],
\end{aligned}
$$

where we use $\frac{B}{n}(1-\gamma_t)^2 + \frac{n-B}{n} \leq 1 - \frac{2B}{n}\gamma_t + \frac{B}{n}\gamma_t^2 \leq 1 - \frac{B}{n}\gamma_t$ due to $\gamma_t < 1$. Second

$$
\begin{aligned}
&\frac{B}{n}\alpha_t^2\,\|g_i(\mathbf{w}_t) - g_i(\mathbf{w}_{t-1})\|_2^2] + \frac{n-B}{n}\|g_i(\mathbf{w}_{t-1}) - g_i(\mathbf{w}_t)\|_2^2 \\
&\leq \left(\frac{B}{n}\frac{(n-B)^2}{B^2(1-\gamma_t)^2} + \frac{n-B}{n}\right)G_2^2\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2 \leq \frac{4n-4B}{B}G_2^2\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2,
\end{aligned}
$$

where we use $\frac{B}{n}\frac{(n-B)^2}{B^2(1-\gamma_t)^2}+\frac{n-B}{n} \leq \frac{n-B}{n}\left(\frac{(n-B)}{B(1-\gamma_t)^2}+1\right) \leq \frac{n-B}{n}\left(\frac{4(n-B)}{B}+4\right) = \frac{4n-4B}{B}$ due to $\gamma_t \leq 1/2$. Third,

$$\begin{aligned}
&\mathbb{E}\left[\frac{B}{n}2\gamma_t'^2\left\|(g_i(\mathbf{w}_t;\zeta_{i,t}) - g_i(\mathbf{w}_{t-1};\zeta_{i,t}) - g_i(\mathbf{w}_t) + g_i(\mathbf{w}_{t-1}))\right\|_2^2\right] \\
&\leq \frac{B}{n}2\gamma_t'^2\mathbb{E}\left[\left\|(g_i(\mathbf{w}_t;\zeta_{i,t}) - g_i(\mathbf{w}_{t-1};\zeta_{i,t})\right\|_2^2\right] \\
&\leq \frac{B}{n}2\left(\frac{n-B}{B(1-\gamma_t)}+1-\gamma_t\right)^2 G_2^2\|\mathbf{w}_t-\mathbf{w}_{t-1}\|_2^2 \leq \frac{8n-4B}{B}G_2^2\|\mathbf{w}_t-\mathbf{w}_{t-1}\|_2^2,
\end{aligned}$$

where we use $\frac{B}{n}2(\frac{n-B}{B(1-\gamma_t)}+1-\gamma_t)^2 \leq \frac{B}{n}2(\frac{2(n-B)}{B}+1)^2 \leq \frac{B}{n}2(\frac{2n-B}{B})^2 = \frac{2(2n-B)(2n-B)}{nB} \leq \frac{8n-4B}{B}$.

Combining the above results, we have

$$\begin{aligned}
\mathbb{E}\left[\|\mathbf{u}_{i,t}-g_i(\mathbf{w}_t)\|_2^2\right] &\leq (1-\frac{B}{n}\gamma_t)\mathbb{E}[\|\mathbf{u}_{i,t-1}-g_i(\mathbf{w}_{t-1})\|_2^2] \\
&+ \frac{12n-8B}{B}G_2^2\|\mathbf{w}_t-\mathbf{w}_{t-1}\|_2^2 + \frac{B}{n}2\gamma_t^2\sigma_0^2.
\end{aligned}$$

Averaging over $i = 1,\ldots,n$ concludes the proof. □

Lemma 5.6 *Consider the $\mathbf{u}_t$ updates in Algorithm 15. Suppose that Assumption 5.1 and 5.3 hold. With $\gamma_t \leq \frac{1}{2}$ and $\gamma_t' = \frac{n-B}{B(1-\gamma_t)} + (1-\gamma_t)$, we have*

$$\mathbb{E}\left[\|\mathbf{u}_t-\mathbf{u}_{t-1}\|_2^2\right] \leq 6B\gamma_t^2\sigma_0^2 + 6B\gamma_t^2\mathbb{E}[\delta_{t-1}] + \frac{10n^2G_2^2}{B}\mathbb{E}\left[\|\mathbf{w}_t-\mathbf{w}_{t-1}\|_2^2\right].$$

Proof. Since $\|\mathbf{u}_t-\mathbf{u}_{t-1}\|_2^2 = \sum_{i=1}^n\|\mathbf{u}_{i,t}-\mathbf{u}_{i,t-1}\|_2^2$, with a probability B/n we have $\mathbf{u}_{i,t} = \bar{\mathbf{u}}_{i,t}$ and a probability $1-B/n$ we have $\mathbf{u}_{i,t} = \mathbf{u}_{i,t-1}$, then

$$\begin{aligned}
&\mathbb{E}\left[\|\mathbf{u}_t-\mathbf{u}_{t-1}\|_2^2\right] \\
&= \frac{B}{n}\sum_{i=1}^n\mathbb{E}\left[\left\|-\gamma_t\mathbf{u}_{i,t-1}+\gamma_t g_i\left(\mathbf{w}_t;\zeta_{i,t}\right)+\gamma_t'\left(g_i\left(\mathbf{w}_t;\zeta_{i,t}\right)-g_i\left(\mathbf{w}_{t-1};\zeta_{i,t}\right)\right)\right\|_2^2\right] \\
&\leq \frac{B}{n}\sum_{i=1}^n\mathbb{E}\left[2\gamma_t^2\left\|g_i\left(\mathbf{w}_t;\zeta_{i,t}\right)-\mathbf{u}_{i,t-1}\right\|_2^2+2(\gamma_t')^2\left\|g_i\left(\mathbf{w}_t;\zeta_{i,t}\right)-g_i\left(\mathbf{w}_{t-1};\zeta_{i,t}\right)\right\|_2^2\right] \\
&\leq \frac{B}{n}\sum_{i=1}^n\mathbb{E}\left[2\gamma_t^2\left\|g_i\left(\mathbf{w}_t;\zeta_{i,t}\right)-\mathbf{u}_{i,t-1}\right\|_2^2\right]+2B(\gamma_t')^2G_2^2\left\|\mathbf{w}_t-\mathbf{w}_{t-1}\right\|_2^2.
\end{aligned}$$

To the first term on the RHS, we use the Young's inequality and Lipschitz continuity of g_i:

$$\begin{aligned}
&\mathbb{E}\left[\left\|g_i\left(\mathbf{w}_t;\zeta_{i,t}\right)-\mathbf{u}_{i,t-1}\right\|_2^2\right] \leq 3\mathbb{E}\left[\left\|g_i\left(\mathbf{w}_t;\zeta_{i,t}\right)-g_i\left(\mathbf{w}_t\right)\right\|_2^2\right] \\
&+3\mathbb{E}\left[\left\|g_i\left(\mathbf{w}_t\right)-g_i\left(\mathbf{w}_{t-1}\right)\right\|_2^2+3\left\|g_i\left(\mathbf{w}_{t-1}\right)-\mathbf{u}_{i,t-1}\right\|_2^2\right] \\
&\leq 3\sigma_0^2+3G_2^2\mathbb{E}\left[\left\|\mathbf{w}_t-\mathbf{w}_{t-1}\right\|_2^2+3\left\|g_i\left(\mathbf{w}_{t-1}\right)-\mathbf{u}_{i,t-1}\right\|_2^2\right].
\end{aligned}$$

Combining the above results, we have

$$\begin{aligned}
&\mathbb{E}\left[\left\|\mathbf{u}_t-\mathbf{u}_{t-1}\right\|_2^2\right] \\
&\leq 6B\gamma_t^2\sigma_0^2+6B\gamma_t^2\mathbb{E}\left[\delta_{t-1}\right]+2BG_2^2(3\gamma_t^2+(\gamma_t')^2)\mathbb{E}\left[\left\|\mathbf{w}_t-\mathbf{w}_{t-1}\right\|_2^2\right].
\end{aligned}$$

With $\gamma_t \leq \frac{1}{2}$, we have $\gamma_t' \leq \frac{2n}{B}$, which yields $(3\gamma_t^2+(\gamma_t')^2) \leq \frac{5n^2}{B^2}$. □

Next, we analyze error recursion of $\Delta_t := \|\mathbf{v}_t - \mathcal{M}_t\|_2^2$.

Lemma 5.7 *Consider the $\mathbf{v}_t$ updates in Algorithm 15 and suppose that Assumption 5.1 and 5.3 hold. Then we have*

$$\begin{aligned}
\mathbb{E}[\Delta_t] &\leq (1-\beta_t)\mathbb{E}[\Delta_{t-1}]+\frac{24G_2^2L_1^2B\gamma_t^2}{n}\mathbb{E}[\delta_{t-1}] \\
&+\left(4L_2^2G_1^2+\frac{40G_2^4L_1^2n}{B}\right)\mathbb{E}\left[\left\|\mathbf{w}_{t-1}-\mathbf{w}_t\right\|_2^2\right]+2\beta_t^2\sigma^2+\frac{24G_2^2L_1^2B}{n}\gamma_t^2\sigma_0^2,
\end{aligned}$$

where $\sigma^2 = \frac{G_1^2\sigma_2^2}{B}+\frac{G_1^2G_2^2}{B}\frac{n-B}{n-1}$.

Proof. Similar to Lemma 5.2, we have $\mathbb{E}_t[\|\mathbf{z}_t - \mathcal{M}_t\|_2^2] \leq \sigma^2$. Since $\mathbf{v}_t = (1-\beta_t)\mathbf{v}_{t-1}+\beta_t\mathbf{z}_t+(1-\beta_t)(\mathbf{z}_t-\tilde{\mathbf{z}}_{t-1})$, applying Lemma 4.11, we have

$$\mathbb{E}_t\left[\left\|\mathbf{v}_t-\mathcal{M}_t\right\|_2^2\right] \leq (1-\beta_t)\left\|\mathbf{v}_{t-1}-\mathcal{M}_{t-1}\right\|_2^2+\mathbb{E}_t[2\|\mathbf{z}_t-\tilde{\mathbf{z}}_{t-1}\|_2^2]+2\beta_t^2\sigma^2.$$

To bound $\mathbb{E}_t[\|\mathbf{z}_t - \tilde{\mathbf{z}}_{t-1}\|_2^2]$, we have

$$\begin{aligned}
&\mathbb{E}_t[\|\mathbf{z}_t-\tilde{\mathbf{z}}_{t-1}\|_2^2] \\
&\leq 2\mathbb{E}_t\left[\frac{1}{B}\sum_{i\in\mathcal{B}_t'}\left\|\nabla g_i(\mathbf{w}_t;\zeta_{i,t}')\nabla f_i(\mathbf{u}_{i,t})-\nabla g_i(\mathbf{w}_t;\zeta_{i,t}')\nabla f_i(\mathbf{u}_{i,t-1})\right\|_2^2\right] \\
&+2\mathbb{E}_t\left[\frac{1}{B}\sum_{i\in\mathcal{B}_t'}\left\|\nabla g_i(\mathbf{w}_t;\zeta_{i,t}')\nabla f_i(\mathbf{u}_{i,t-1})-\nabla g_i(\mathbf{w}_{t-1};\zeta_{i,t}')\nabla f_i(\mathbf{u}_{i,t-1})\right\|_2^2\right] \\
&\leq 2G_2^2L_1^2\mathbb{E}_t\left[\frac{1}{B}\sum_{i\in\mathcal{B}_t'}\|\mathbf{u}_{i,t}-\mathbf{u}_{i,t-1}\|_2^2\right]+2L_2^2G_1^2\|\mathbf{w}_t-\mathbf{w}_{t-1}\|_2^2 \\
&= 2G_2^2L_1^2\mathbb{E}_t\left[\frac{1}{n}\sum_{i=1}^{n}\|\mathbf{u}_{i,t}-\mathbf{u}_{i,t-1}\|_2^2\right]+2L_2^2G_1^2\|\mathbf{w}_t-\mathbf{w}_{t-1}\|_2^2,
\end{aligned}$$

where the last inequality follows the Assumption 5.3.

As a result, we have

$$\mathbb{E}[\Delta_t] \leq (1-\beta_t)\mathbb{E}[\Delta_{t-1}] + \frac{4G_2^2L_1^2}{n}\mathbb{E}\left[\|\mathbf{u}_t - \mathbf{u}_{t-1}\|_2^2\right] + 4L_2^2G_1^2\mathbb{E}\left[\|\mathbf{w}_{t-1} - \mathbf{w}_t\|_2^2\right] + 2\beta_t^2\sigma^2.$$

Combining with the result in Lemma 5.6, i.e.,

$$\mathbb{E}\left[\|\mathbf{u}_t - \mathbf{u}_{t-1}\|_2^2\right] \leq 6B\gamma_t^2\sigma_0^2 + 6B\gamma_t^2\mathbb{E}[\delta_{t-1}] + \frac{10n^2G_2^2}{B}\mathbb{E}\left[\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2\right].$$

we have

$$\begin{aligned}\mathbb{E}[\Delta_t] \leq\ & (1-\beta_t)\mathbb{E}[\Delta_{t-1}] + \frac{24BG_2^2L_1^2}{n}\gamma_t^2\mathbb{E}[\delta_{t-1}] \\ & + \left(4L_2^2G_1^2 + \frac{40G_2^4L_1^2n}{B}\right)\mathbb{E}\left[\|\mathbf{w}_{t-1} - \mathbf{w}_t\|_2^2\right] + 2\beta_t^2\sigma^2 + \frac{24BG_2^2L_1^2\gamma_t^2\sigma_0^2}{n},\end{aligned}$$

which completes the proof. □

Lemma 5.8 *For the update $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t\mathbf{v}_t, t \geq 0$, if $\eta_t \leq 1/(2L_F)$ we have*

$$F(\mathbf{w}_{t+1}) \leq F(\mathbf{w}_t) + G_2^2L_1^2\eta_t\delta_t + \eta_t\Delta_t - \frac{\eta_t}{2}\|\nabla F(\mathbf{w}_t)\|_2^2 - \frac{1}{4\eta_t}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2. \tag{5.13}$$

Proof. It follows directly from Lemma 4.9 by noting that

$$\begin{aligned}&\|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|_2^2 = \|\mathbf{v}_t - \mathcal{M}_t + \mathcal{M}_t - \nabla F(\mathbf{w}_t)\|_2^2 \\ &\leq 2\Delta_t + 2\left\|\frac{1}{n}\sum_{i=1}^n \nabla g_i(\mathbf{w}_t)\nabla f_i(\mathbf{u}_{i,t}) - \frac{1}{n}\sum_{i=1}^n \nabla g_i(\mathbf{w}_t)\nabla f_i(g_i(\mathbf{w}_t))\right\|_2^2 \\ &\leq 2\Delta_t + \frac{2G_2^2L_1^2}{n}\sum_{i=1}^n \|\mathbf{u}_{i,t} - g_i(\mathbf{w}_t)\|_2^2.\end{aligned}$$

Taking expectation over all randomness on both sides yields the desired result. □

Now we state the convergence theorem for MSVR.

Theorem 5.2 *Suppose that Assumption 5.1 and 5.3 hold. Let $\beta = O(\frac{\epsilon\eta L_1\sqrt{n}}{\sigma\sqrt{B}})$, $\gamma = \min\left(\frac{\epsilon\eta L_1 n}{\sigma_0 B}, 1\right)$, $\eta = \min\left(\frac{1}{2L_F}, O(\frac{\epsilon\sqrt{B}}{L_1\sigma\sqrt{n}}), O(\frac{\epsilon B}{L_1^2\sigma_0 n}), O(\frac{B}{nL_1})\right)$. Then MSVR can find $\mathbf{w}_\tau$ that is sampled randomly from $\{0, \ldots, T-1\}$ satisfying*

$$\mathbb{E}\left[\|\mathbf{v}_\tau\|_2^2 + \|\nabla F(\mathbf{w}_\tau)\|_2^2\right] \leq O(\epsilon).$$

with an iteration complexity of

$$T = O\left(\max\left\{\frac{C_\Upsilon L_F}{\epsilon^2}, \frac{C_\Upsilon L_1 n}{\epsilon^2 B}, \frac{C_\Upsilon L_1 \sigma\sqrt{n}}{\epsilon^3\sqrt{B}}, \frac{C_\Upsilon L_1^2 \sigma_0 n}{\epsilon^3 B}\right\}\right).$$

where $\sigma^2 = \frac{G_1^2\sigma_2^2}{B} + \frac{G_1^2 G_2^2(n-B)}{B(n-1)}$, $C_\Upsilon = O(F(\mathbf{w}_0) - F_* + \frac{B}{nL_1^2\eta}\Delta_0 + \frac{B}{nL_1^2\eta}\delta_0)$.

Why it matters

Theorem 5.2 indicates that when the initial estimators $\mathbf{u}_0$ and $\mathbf{v}_0$ have an estimation error in the order of $O(\epsilon)$ such that C_Υ is $O(1)$, MSVR attains a better complexity than SOX for finding an ϵ-stationary solution under stronger assumptions of the mean-Lipschitz continuity of g and ∇g. Its complexity is comparable to that of SCST in Theorem 4.4, up to a factor of n/B.

Proof. We have established the following:

$$(*)\ F(\mathbf{w}_{t+1}) \leq F(\mathbf{w}_t) + G_2^2 L_1^2 \eta_t \delta_t + \eta_t \Delta_t - \frac{\eta_t}{2}\|\nabla F(\mathbf{w}_t)\|_2^2 - \frac{1}{4\eta_t}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2$$

$$(\sharp)\mathbb{E}[\Delta_t] \leq (1-\beta_t)\mathbb{E}[\Delta_{t-1}] + \frac{24BG_2^2L_1^2}{n}\gamma_t^2\mathbb{E}[\delta_{t-1}]$$
$$+ \left(4L_2^2G_1^2 + \frac{40G_2^4L_1^2 n}{B}\right)\mathbb{E}\left[\|\mathbf{w}_{t-1} - \mathbf{w}_t\|_2^2\right] + 2\beta_t^2\sigma^2 + \frac{24BG_2^2L_1^2\sigma_0^2}{n}\gamma_t^2,$$

$$(\diamond)\mathbb{E}\left[\delta_t\right] \leq \left(1 - \frac{B\gamma_t}{n}\right)\mathbb{E}\left[\delta_{t-1}\right] + \frac{2B}{n}\gamma_t^2\sigma_0^2 + \frac{12nG_2^2}{B}\mathbb{E}\left[\|\mathbf{w}_t - \mathbf{w}_{t-1}\|^2\right].$$

In order to apply Lemma 4.15, we let $A_t = F(\mathbf{w}_t) - F_*$, $B_t = \|\nabla F(\mathbf{w}_t)\|_2^2/2$, $\Gamma_t = \|\mathbf{v}_t\|_2^2/4$, $\bar{\delta}_t = L_1^2G_2^2\delta_t$, $\bar{\gamma}_t = \frac{B\gamma_t}{n}$. Then the following three inequalities

$$(*)\mathbb{E}[A_{t+1}] \leq \mathbb{E}[A_t + \eta_t\Delta_t + \eta_t\bar{\delta}_t - \eta_t B_t - \eta_t\Gamma_t]$$
$$(\sharp)\mathbb{E}\left[\Delta_{t+1}\right] \leq \mathbb{E}[(1-\beta_{t+1})\Delta_t + C_1\bar{\gamma}_{t+1}^2\bar{\delta}_t + C_2\eta_t^2\Gamma_t + \beta_{t+1}^2\sigma^2 + \bar{\gamma}_{t+1}^2\sigma'^2],$$
$$(\diamond)\mathbb{E}\left[\bar{\delta}_{t+1}\right] \leq \mathbb{E}[(1-\bar{\gamma}_{t+1})\bar{\delta}_t + C_3\eta_t^2\Gamma_t + \bar{\gamma}_{t+1}^2\sigma''^2].$$

hold with $C_1 = O(n/B), C_2 = O(L_1^2 n/B + L_2^2), C_3 = O(L_1^2 n/B), \sigma^2 = \frac{G_1^2\sigma_2^2}{B} + \frac{G_1^2G_2^2(n-B)}{B(n-1)}), \sigma'^2 = O(L_1^2\sigma_0^2 n/B), \sigma''^2 = O(L_1^2\sigma_0^2 n/B)$. Following the settings in Lemma 4.15, we can finish the proof with

	f_i			g_i			F
	Lipschitz continuity	Weak convexity	Monotonicity	Lipschitz continuity	Weak convexity	Smoothness	Weak convexity (ρ)
5.5(i)	G_1	ρ_1	$\partial f \geq 0$	G_2	ρ_2	-	$G_1\rho_2\sqrt{d'} + \rho_1 G_2^2$
5.5(ii)	G_1	ρ_1	$\partial f \geq 0$ or $\partial f \leq 0$	G_2	-	L_2	$G_1 L_2\sqrt{d'} + \rho_1 G_2^2$

Table 5.1: Conditions of f_i and g_i to make $F(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^n f_i(g_i(\mathbf{w}))$ weakly convex, where $g_i : \mathbb{R}^d \to \mathbb{R}^{d'}$ and $f_i : \mathbb{R}^{d'} \to \mathbb{R}$.

$$
\begin{aligned}
\eta &= \min\left(\frac{1}{L}, \frac{\epsilon}{4\sqrt{C_2}\sigma}, \frac{\epsilon\sqrt{C_2}}{8C_3\sigma'}, \frac{\epsilon}{8\sqrt{C_3}\sigma''}, \frac{\sqrt{C_2}}{4C_3\sqrt{C_1}}\right) \\
&= \min\left(\frac{1}{2L_F}, O\left(\frac{\epsilon}{L_1\sigma}\sqrt{\frac{B}{n}}\right), O\left(\frac{\epsilon B}{L_1^2\sigma_0 n}\right), O\left(\frac{B}{nL_1}\right)\right), \\
\beta &= \frac{\epsilon\eta\sqrt{2C_2}}{2\sigma} = O\left(\frac{\epsilon\eta L_1}{2\sigma}\sqrt{\frac{n}{B}}\right), \\
\tilde{\gamma} &= \min\left(\frac{\epsilon\eta\sqrt{C_2}}{\sigma'}, \frac{\epsilon\eta\sqrt{C_3}}{\sigma''}, \frac{C_2}{2C_3C_1}\right) = \min\left(O\left(\frac{\epsilon\eta}{\sigma_0}\right), O\left(\frac{B}{n}\right)\right), \\
T &= O\left(\max\left\{\frac{C_\Upsilon L_F}{\epsilon^2}, \frac{C_\Upsilon L_1 n}{\epsilon^2 B}, \frac{C_\Upsilon L_1\sigma\sqrt{n}}{\epsilon^3\sqrt{B}}, \frac{C_\Upsilon L_1^2\sigma_0 n}{\epsilon^3 B}\right\}\right).
\end{aligned}
$$

where $C_\Upsilon = F(\mathbf{w}_0) - F_* + \frac{1}{4C_2\eta}\Delta_0 + \frac{1}{4C_3\eta}\delta_0$.

□

5.3 Non-Smooth Weakly Convex Functions

In this section, we consider non-smooth weakly convex functions, where either the outer function or the inner function are non-smooth. The group DRO objective (5.2) falls into this category. Another instance is the two-way partial AUC maximization problem as discussed in Section 6.4.3.

Assumption 5.4. *We assume that*

(i) $\mathbb{E}_{\zeta\sim\mathbb{P}_i}[\|g_i(\mathbf{w};\zeta) - g_i(\mathbf{w})\|_2^2] \leq \sigma_0^2$.

(ii) $\mathbb{E}_{\zeta\sim\mathbb{P}_i}[\|\mathcal{G}_i(\mathbf{w};\zeta)\|_2^2] \leq G_2^2$ *for any* $\mathcal{G}_i(\mathbf{w};\zeta) \in \partial g_i(\mathbf{w};\zeta)$.

The second condition above implies that g_i is G_2-Lipschitz continuous.

Assumption 5.5. *We assume either of the following conditions holds:*

(i) f_i *is* ρ_1*-weakly convex,* G_1*-Lipschitz continuous, and* $\partial f_i(g) \geq 0\ \forall g$*;* g_i *is* ρ_2*-weakly convex.*

(ii) f_i is ρ_1-weakly convex, G_1-Lipschitz continuous, and $\partial f_i(g) \geq 0$ or $\partial f_i(g) \leq 0$ $\forall g$; and g_i is L_2-smooth.

We first characterize the conditions on f_i and g_i to induce weak convexity of F.

Lemma 5.9 *Under Assumption 5.4 and 5.5, the objective function F is ρ-weakly convex for some $\rho > 0$. If Assumption 5.5(i) holds, then $\rho = G_1\rho_2\sqrt{d'} + \rho_1 G_2^2$ and if Assumption 5.5(ii) holds, then $\rho = G_1 L_2\sqrt{d'} + \rho_1 G_2^2$.*

Proof. The weak convexity of f_i implies that for any $\mathbf{v}_i \in \partial f_i(g_i(\mathbf{w}))$:

$$\begin{aligned} f_i(g_i(\mathbf{w}')) &\geq f_i(g_i(\mathbf{w})) + \mathbf{v}_i^\top(g_i(\mathbf{w}') - g_i(\mathbf{w})) - \frac{\rho_1}{2}\|g_i(\mathbf{w}') - g_i(\mathbf{w})\|_2^2 \\ &\geq f_i(g_i(\mathbf{w})) + \mathbf{v}_i^\top(g_i(\mathbf{w}') - g_i(\mathbf{w})) - \frac{\rho_1 G_2^2}{2}\|\mathbf{w} - \mathbf{w}'\|_2^2. \end{aligned}$$

Let us first prove the weak convexity under Assumption 5.5(i). Since g_i is ρ_2-weakly convex, we have for any $U_i \in \partial g_i(\mathbf{w})$

$$g_i(\mathbf{w}') - g_i(\mathbf{w}) \geq U_i^\top(\mathbf{w}' - \mathbf{w}) - \frac{\rho_2}{2}\|\mathbf{w}' - \mathbf{w}\|_2^2\mathbf{1}. \tag{5.14}$$

where $\mathbf{1}$ denotes a vector of all ones. Since $\mathbf{v}_i \geq 0$, we have

$$\begin{aligned} f_i(g_i(\mathbf{w}')) &\geq f_i(g_i(\mathbf{w})) + \mathbf{v}_i^\top(U_i^\top(\mathbf{w}' - \mathbf{w}) - \frac{\rho_2}{2}\|\mathbf{w} - \mathbf{w}'\|_2^2\mathbf{1}) - \frac{\rho_1 G_2^2}{2}\|\mathbf{w} - \mathbf{w}'\|_2^2 \\ &\geq f_i(g_i(\mathbf{w})) + (U_i\mathbf{v}_i)^\top(\mathbf{w}' - \mathbf{w}) - \frac{G_1\sqrt{d'}\rho_2 + \rho_1 G_2^2}{2}\|\mathbf{w} - \mathbf{w}'\|_2^2 \end{aligned}$$

Since $U_i\mathbf{v}_i \in \partial g_i(\mathbf{w})\partial f_i(g_i(\mathbf{w}))$, the above inequality indicates that $f_i(g_i(\mathbf{w}))$ is ρ-weakly convex, where $\rho = G_1\sqrt{d'}\rho_2 + \rho_1 G_2^2$. As a result, $F(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^n f_i(g_i(\mathbf{w}))$ is ρ-weakly convex.

Next, we prove the weak convexity of F under Assumption 5.5(ii). Due to the smoothness of $g(\cdot)$ we have

$$\begin{aligned} g(\mathbf{w}) - g(\mathbf{w}') &\leq \nabla g(\mathbf{w}')^\top(\mathbf{w} - \mathbf{w}') + \frac{L_2}{2}\|\mathbf{w} - \mathbf{w}'\|_2^2\mathbf{1}, \\ g(\mathbf{w}) - g(\mathbf{w}') &\geq \nabla g(\mathbf{w}')^\top(\mathbf{w} - \mathbf{w}') - \frac{L_2}{2}\|\mathbf{w} - \mathbf{w}'\|_2^2\mathbf{1}. \end{aligned} \tag{5.15}$$

If $\partial f_i(g_i(\mathbf{w})) \geq 0$, we use the second inequity above and follow the same steps as before to prove the ρ-weak convexity of F with $\rho = G_1\sqrt{d'}L_2 + \rho_1 G_2^2$. If $\partial f_i(g_i(\mathbf{w})) \leq 0$, we will use the first inequality above to get:

Algorithm 16 SONX

Input: learning rate schedules $\{\eta_t\}_{t=1}^T$, $\{\gamma_t\}_{t=1}^T$, $\{\beta_t\}_{t=1}^T$; starting points $\mathbf{w}_0$, $\mathbf{u}_0$
$\mathbf{w}_1 = \mathbf{w}_0$
for $t = 1, \ldots, T$ **do**
  Draw a batch of samples $\mathcal{B}_t \subset [n]$
  for $i \in \mathcal{B}_t$ **do**
    Draw two samples $\zeta_{i,t} \sim \mathbb{P}_i$
    Update the inner function value estimators by

$$\text{v1:} \quad \mathbf{u}_{i,t} = (1-\gamma_t)\mathbf{u}_{i,t-1} + \gamma_t g_i(\mathbf{w}_t; \zeta_{i,t})$$
$$\text{v2:} \quad \mathbf{u}_{i,t} = (1-\gamma_t)\mathbf{u}_{i,t-1} + \gamma_t g_i\left(\mathbf{w}_t; \zeta_{i,t}\right) + \gamma_t' \left(g_i\left(\mathbf{w}_t; \zeta_{i,t}\right) - g_i\left(\mathbf{w}_{t-1}; \zeta_{i,t}\right)\right)$$

  end for
  Set $\mathbf{u}_{i,t} = \mathbf{u}_{i,t-1}$, $i \notin \mathcal{B}_t$
  Compute $\mathbf{z}_t = \frac{1}{B}\sum_{i\in\mathcal{B}_t'} \partial g_i(\mathbf{w}_t; \zeta_{i,t}')\partial f_i(\mathbf{u}_{i,t})$ ◇ check text for discussion
  Update the model $\mathbf{w}$ by $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{z}_t$
end for

$$\begin{aligned}
f_i(g_i(\mathbf{w}')) &\geq f_i(g_i(\mathbf{w})) + \mathbf{v}_i^\top (g_i(\mathbf{w}') - g_i(\mathbf{w})) - \frac{\rho_1}{2}\|g_i(\mathbf{w}') - g_i(\mathbf{w})\|_2^2 \\
&\geq f_i(g_i(\mathbf{w})) + \mathbf{v}_i^\top (\nabla g_i(\mathbf{w})^\top (\mathbf{w}' - \mathbf{w}) + \frac{L_2}{2}\|\mathbf{w} - \mathbf{w}'\|_2^2 \mathbf{1}) - \frac{\rho_1 G_2^2}{2}\|\mathbf{w} - \mathbf{w}'\|_2^2 \\
&\geq f_i(g_i(\mathbf{w})) + (\nabla g_i(\mathbf{w})\mathbf{v}_i)^\top (\mathbf{w}' - \mathbf{w}) - \frac{G_1\sqrt{d'}L_2 + \rho_1 G_2^2}{2}\|\mathbf{w} - \mathbf{w}'\|_2^2.
\end{aligned}$$

This concludes the proof. □

5.3.1 SONX for Non-smooth Inner Functions

Since we do not assume smoothness for the overall objective function, the key difference from the previous two sections is that we no longer have the descent lemma in Lemma 4.9, hence cannot leverage the MA or STORM gradient estimators. Consequently, we employ the vanilla gradient estimator $\mathbf{z}_t$ to update the model parameter $\mathbf{w}_{t+1}$. The updating steps are summarized in Algorithm 16, referred to as **SONX**. The two options correspond to different strategies for updating the inner function value estimators: v1 uses a coordinate MA estimator, while v2 adopts the MSVR estimator.

For ease of presentation, we compute the vanilla gradient estimator $\mathbf{z}_t$ using a batch $\mathcal{B}_t'$ independent from $\mathcal{B}_t$:

$$\mathbf{z}_t = \frac{1}{B}\sum_{i\in\mathcal{B}_t'} \partial g_i(\mathbf{w}_t; \zeta_{i,t}')\partial f_i(\mathbf{u}_{i,t}).$$

However, for SONX-v1 with MA estimator, we can indeed use the same vanilla gradient estimator $\mathbf{z}_t$ as in SOX:

$$\mathbf{z}_t = \frac{1}{B}\sum_{i\in\mathcal{B}_t} \partial g_i(\mathbf{w}_t;\zeta'_{i,t})\partial f_i(\mathbf{u}_{i,t}).$$

An alternative method for using both options is to compute $\mathbf{z}_t$ by

$$\mathbf{z}_t = \frac{1}{B}\sum_{i\in\mathcal{B}_t} \partial g_i(\mathbf{w}_t;\zeta'_{i,t})\partial f_i(\mathbf{u}_{i,t-1}).$$

Convergence Analysis

Similar to Section 3.1.4, we state the convergence using the Moreau envelope of F:

$$F_\lambda(\mathbf{w}) := \min_{\mathbf{u}} F(\mathbf{u}) + \frac{1}{2\lambda}\|\mathbf{u}-\mathbf{w}\|_2^2.$$

Recall the definition:

$$\mathrm{prox}_{\lambda F}(\mathbf{w}) = \arg\min_{\mathbf{u}} F(\mathbf{u}) + \frac{1}{2\lambda}\|\mathbf{u}-\mathbf{w}\|_2^2.$$

We first present a result similar to Lemma 3.5 for standard SGD to account for the bias of $\mathbf{z}_t$.

Lemma 5.10 *Suppose Assumption 5.4 and 5.5 hold. Let $\bar\rho = \rho + \rho_2 G_1 + 2\rho_1 G_2^2$. Consider the step update of SONX, we have*

$$\mathbb{E}_{\zeta'_t,\mathcal{B}'_t}[F_{1/\bar\rho}(\mathbf{w}_{t+1})] \le F_{1/\bar\rho}(\mathbf{w}_t) + \frac{\eta_t^2\bar\rho G^2}{2} - \frac{\eta_t}{2}\|\nabla F_{1/\bar\rho}(\mathbf{w}_t)\|_2^2$$
$$+ \frac{\bar\rho\eta_t}{n}\sum_{i=1}^n\left[2G_1\|g_i(\mathbf{w}_t)-\mathbf{u}_{i,t}\|_2 + \rho_1\|g_i(\mathbf{w}_t)-\mathbf{u}_{i,t}\|_2^2\right].$$

If f_i is further L_1-smooth, then

$$\mathbb{E}_{\zeta'_t,\mathcal{B}'_t}[F_{1/\bar\rho}(\mathbf{w}_{t+1})] \le F_{1/\bar\rho}(\mathbf{w}_t) + \frac{\eta_t^2\bar\rho G^2}{2} - \frac{\eta_t}{2}\|\nabla F_{1/\bar\rho}(\mathbf{w}_t)\|_2^2$$
$$+ \frac{\bar\rho\eta_t}{n}\sum_{i=1}^n\left[\frac{L_1}{2}\|g_i(\mathbf{w}_t)-\mathbf{u}_{i,t}\|_2^2 + \rho_1\|g_i(\mathbf{w}_t)-\mathbf{u}_{i,t}\|_2^2\right].$$

where $G^2 = G_1^2G_2^2$.

If $\mathbf{u}_{i,t} = g_i(\mathbf{w}_t)$, i.e., there is no bias in $\mathbf{z}_t$, then the terms in the square bracket are gone, the above lemma reduces to Lemma 3.4.

Proof. Define $\hat{\mathbf{w}}_t := \mathrm{prox}_{F/\bar\rho}(\mathbf{w}_t)$ and $\mathbb{E}_t[\cdot] = \mathbb{E}_{\zeta'_t,\mathcal{B}'_t}[\cdot]$. First,

$$\mathbb{E}_t[\|\mathbf{z}_t\|_2^2] \le \mathbb{E}_t\left[\frac{1}{B}\sum_{i\in\mathcal{B}_t}\|\partial g_i(\mathbf{w}_t,\zeta'_{i,t})\partial f_i(\mathbf{u}_{i,t})\|_2^2\right]$$
$$\le \mathbb{E}_t\left[\frac{1}{B}\sum_{i\in\mathcal{B}_t}\|\partial g_i(\mathbf{w}_t,\zeta'_{i,t})\|_2^2 G_1^2\right] \le G_2^2 G_1^2 = G^2.$$

Following Lemma 3.4, we have

$$\mathbb{E}_t[F_{1/\bar\rho}(\mathbf{w}_{t+1})] \le F_{1/\bar\rho}(\mathbf{w}_t) + \bar\rho\eta_t(\mathbb{E}_t[\mathbf{z}_t])^\top(\hat{\mathbf{w}}_t - \mathbf{w}_t) + \frac{\eta_t^2\bar\rho G^2}{2}. \tag{5.16}$$

Next we bound the term $\mathbb{E}_t[\mathbf{z}_t]^\top(\hat{\mathbf{w}}_t - \mathbf{w}_t)$ on the RHS of (5.16). Note that $\mathbb{E}_t[\mathbf{z}_t] = \frac{1}{n}\sum_{i=1}^n \partial g_i(\mathbf{w}_t)\partial f_i(\mathbf{u}_{i,t})$. For a given $i \in [n]$, we have

$$\begin{aligned}
&f_i(g_i(\hat{\mathbf{w}}_t)) - f_i(\mathbf{u}_{i,t})\\
&\overset{(a)}{\ge} \partial f_i(\mathbf{u}_{i,t})^\top(g_i(\hat{\mathbf{w}}_t) - \mathbf{u}_{i,t}) - \frac{\rho_1}{2}\|g_i(\hat{\mathbf{w}}_t) - \mathbf{u}_{i,t}\|_2^2\\
&\ge \partial f_i(\mathbf{u}_{i,t})^\top(g_i(\hat{\mathbf{w}}_t) - \mathbf{u}_{i,t}) - \rho_1\|g_i(\hat{\mathbf{w}}_t) - g_i(\mathbf{w}_t)\|_2^2 - \rho_1\|g_i(\mathbf{w}_t) - \mathbf{u}_{i,t}\|_2^2\\
&\ge \partial f_i(\mathbf{u}_{i,t})^\top(g_i(\hat{\mathbf{w}}_t) - \mathbf{u}_{i,t}) - \rho_1 G_2^2\|\hat{\mathbf{w}}_t - \mathbf{w}_t\|_2^2 - \rho_1\|g_i(\mathbf{w}_t) - \mathbf{u}_{i,t}\|_2^2\\
&\overset{(b)}{\ge} \partial f_i(\mathbf{u}_{i,t})^\top\left[g_i(\mathbf{w}_t) - \mathbf{u}_{i,t} + \partial g_i(\mathbf{w}_t)^\top(\hat{\mathbf{w}}_t - \mathbf{w}_t) - \frac{\rho_2}{2}\|\hat{\mathbf{w}}_t - \mathbf{w}_t\|_2^2\right]\\
&\quad - \rho_1 G_2^2\|\hat{\mathbf{w}}_t - \mathbf{w}_t\|_2^2 - \rho_1\|g_i(\mathbf{w}_t) - \mathbf{u}_{i,t}\|_2^2\\
&\overset{(c)}{\ge} \partial f_i(\mathbf{u}_{i,t})^\top(g_i(\mathbf{w}_t) - \mathbf{u}_{i,t}) + \partial f_i(\mathbf{u}_{i,t})^\top\partial g_i(\mathbf{w}_t)^\top(\hat{\mathbf{w}}_t - \mathbf{w}_t)\\
&\quad - (\frac{\rho_2 G_1}{2} + \rho_1 G_2^2)\|\hat{\mathbf{w}}_t - \mathbf{w}_t\|_2^2 - \rho_1\|g_i(\mathbf{w}_t) - \mathbf{u}_{i,t}\|_2^2,
\end{aligned}$$

where (a) follows from the ρ_1-weak-convexity of f_i, (b) follows from that $\partial f_i(\cdot) \ge 0$ and the weak convexity of g_i, (c) is due to $\|\partial f_i(\mathbf{u}_{i,t})\|_2 \le G_1$. When $\partial f_i(\cdot) \le 0$ and g_i is smooth, we can bound similarly with ρ_2 in the last inequality replaced by L_2.

Then rearranging the above inequality and averaging over i yields

$$\begin{aligned}
&\mathbb{E}_t[\mathbf{z}_t]^\top(\hat{\mathbf{w}}_t - \mathbf{w}_t) = \frac{1}{n}\sum_{i=1}^n \partial f_i(\mathbf{u}_{i,t})^\top\partial g_i(\mathbf{w}_t)^\top(\hat{\mathbf{w}}_t - \mathbf{w}_t)\\
&\le \frac{1}{n}\sum_{i=1}^n\Big[f_i(g_i(\hat{\mathbf{w}}_t)) - f_i(g_i(\mathbf{w}_t)) + f_i(g_i(\mathbf{w}_t)) - f_i(\mathbf{u}_{i,t})\\
&\quad - \partial f_i(\mathbf{u}_{i,t})^\top(g_i(\mathbf{w}_t) - \mathbf{u}_{i,t}) + (\frac{\rho_2 G_1}{2} + \rho_1 G_2^2)\|\hat{\mathbf{w}}_t - \mathbf{w}_t\|_2^2 + \rho_1\|g_i(\mathbf{w}_t) - \mathbf{u}_{i,t}\|_2^2\Big].
\end{aligned} \tag{5.17}$$

Due to the ρ-weak convexity of $F(\mathbf{w})$, we have that $F(\mathbf{w}) + \frac{\bar\rho}{2}\|\mathbf{w}_t - \mathbf{w}\|_2^2$ is $(\bar\rho - \rho)$-strongly convex. Then $\left[F(\mathbf{w}_t) + \frac{\bar\rho}{2}\|\mathbf{w}_t - \mathbf{w}_t\|_2^2\right] - \left[F(\hat{\mathbf{w}}_t) + \frac{\bar\rho}{2}\|\mathbf{w}_t - \hat{\mathbf{w}}_t\|_2^2\right] \ge \frac{\bar\rho-\rho}{2}\|\hat{\mathbf{w}}_t - \mathbf{w}_t\|_2^2$. It follows that:

$$\begin{aligned}
&\frac{1}{n}\sum_{i=1}^{n}\Big[f_i(g_i(\hat{\mathbf{w}}_t)) - f_i(g_i(\mathbf{w}_t))\Big] = F(\hat{\mathbf{w}}_t) - F(\mathbf{w}_t) \\
&= \Big[F(\hat{\mathbf{w}}_t) + \frac{\bar{\rho}}{2}\|\mathbf{w}_t - \hat{\mathbf{w}}_t\|_2^2\Big] - \Big[F(\mathbf{w}_t) + \frac{\bar{\rho}}{2}\|\mathbf{w}_t - \mathbf{w}_t\|_2^2\Big] - \frac{\bar{\rho}}{2}\|\mathbf{w}_t - \hat{\mathbf{w}}_t\|_2^2 \qquad (5.18)\\
&\leq (\frac{\rho}{2} - \bar{\rho})\|\mathbf{w}_t - \hat{\mathbf{w}}_t\|_2^2
\end{aligned}$$

Combining inequality (5.17), (5.16) and (5.18) yields

$$\begin{aligned}
\mathbb{E}_t[F_{1/\bar{\rho}}(\mathbf{w}_{t+1})] \leq F_{1/\bar{\rho}}(\mathbf{w}_t) + \frac{\eta_t^2\bar{\rho}G^2}{2} - \frac{\bar{\rho}^2\eta_t}{2}\|\mathbf{w}_t - \hat{\mathbf{w}}_t\|_2^2 \\
+ \frac{\bar{\rho}\eta_t}{n}\sum_{i=1}^{n}\Big[f_i(g_i(\mathbf{w}_t)) - f_i(\mathbf{u}_{i,t}) - \partial f_i(\mathbf{u}_{i,t})^\top(g_i(\mathbf{w}_t) - \mathbf{u}_{i,t}) \\
+ \rho_1\|g_i(\mathbf{w}_t) - \mathbf{u}_{i,t}\|_2^2\Big].
\end{aligned}$$

We finish the proof by noting that $\|\nabla F_{1/\bar{\rho}}(\mathbf{w}_t)\|_2 = \bar{\rho}\|\mathbf{w}_t - \hat{\mathbf{w}}_t\|_2$, using

$$f_i(g_i(\mathbf{w}_t)) - f_i(\mathbf{u}_{i,t}) - \partial f_i(\mathbf{u}_{i,t})^\top(g_i(\mathbf{w}_t) - \mathbf{u}_{i,t}) \leq 2G_1\|g_i(\mathbf{w}_t) - \mathbf{u}_{i,t}\|_2,$$

if f_i is G_1-Lipschitz continuous, or using

$$f_i(g_i(\mathbf{w}_t)) - f_i(\mathbf{u}_{i,t}) - \partial f_i(\mathbf{u}_{i,t})^\top(g_i(\mathbf{w}_t) - \mathbf{u}_{i,t}) \leq \frac{L_1}{2}\|g_i(\mathbf{w}_t) - \mathbf{u}_{i,t}\|_2^2,$$

if f_i is L_1-smooth. □

Convergence of SONX-v1

Recall the definition:

$$\delta_t = \frac{1}{n}\sum_{i=1}^{n}\|\mathbf{u}_{i,t} - g_i(\mathbf{w}_t)\|_2^2.$$

Let us also define:

$$\delta_t' = \frac{1}{n}\sum_{i=1}^{n}\|\mathbf{u}_{i,t} - g_i(\mathbf{w}_t)\|_2.$$

From Lemma 5.10, the key is to bound δ_t and δ_t'.

Lemma 5.11 *Consider the update of SONX-v1, under Assumptions 5.4 and 5.5, with constant parameters $\gamma_t = \gamma \leq 1$ and $\eta_t = \eta$, we have*

$$\mathbb{E}\left[\delta_t\right] \leq \left(1-\frac{B\gamma}{4n}\right)^{2t}\delta_0 + \frac{8n^2G_2^4G_1^2\eta^2}{B^2\gamma^2} + 4\gamma\sigma_0^2.$$
$$\mathbb{E}\left[\delta_t'\right] \leq \left(1-\frac{B\gamma}{4n}\right)^{t}\delta_0' + \frac{4nG_2^2G_1\eta}{B\gamma} + 2\sqrt{\gamma}\sigma_0.$$

Proof. From the proof of Lemma 5.1, we have

$$\begin{aligned}
&\mathbb{E}\left[\left\|\mathbf{u}_{i,t} - g_i(\mathbf{w}_t)\right\|_2^2\right] \\
&\leq \left(1-\frac{B\gamma_t}{2n}\right)\mathbb{E}\left[\left\|\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_{t-1})\right\|_2^2\right] + \frac{2nG_2^2}{B\gamma_t}\mathbb{E}\left[\|\mathbf{w}_{t-1}-\mathbf{w}_t\|_2^2\right] + \frac{B\gamma_t^2\sigma_0^2}{n} \\
&\leq \left(1-\frac{B\gamma_t}{2n}\right)\mathbb{E}\left[\left\|\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_{t-1})\right\|_2^2\right] + \frac{2nG_2^2\eta_{t-1}^2}{B\gamma_t}\mathbb{E}\left[\|\mathbf{z}_{t-1}\|_2^2\right] + \frac{B\gamma_t^2\sigma_0^2}{n} \\
&\leq \left(1-\frac{B\gamma_t}{4n}\right)^2\mathbb{E}\left[\left\|\mathbf{u}_{i,t-1} - g_i(\mathbf{w}_{t-1})\right\|_2^2\right] + \frac{2nG_2^4G_1^2\eta_{t-1}^2}{B\gamma_t} + \frac{B\gamma_t^2\sigma_0^2}{n}.
\end{aligned}$$

Applying the above inequality recursively for $\gamma_t = \gamma$ and $\eta_t = \eta$, we obtain

$$\begin{aligned}
&\mathbb{E}\left[\left\|\mathbf{u}_{i,t} - g_i(\mathbf{w}_t)\right\|_2^2\right] \\
&\leq \left(1-\frac{B\gamma}{4n}\right)^{2t}\left\|\mathbf{u}_{i,0} - g_i(\mathbf{w}_0)\right\|_2^2 + \sum_{j=0}^{t-1}\left(1-\frac{B\gamma}{4n}\right)^{2j}\left(\frac{2nG_2^4G_1^2\eta^2}{B\gamma} + \frac{B\gamma^2\sigma_0^2}{n}\right) \\
&\leq \left(1-\frac{B\gamma}{4n}\right)^{2t}\left\|\mathbf{u}_{i,0} - g_i(\mathbf{w}_0)\right\|_2^2 + \frac{8n^2G_2^4G_1^2\eta^2}{B^2\gamma^2} + 4\gamma\sigma_0^2,
\end{aligned}$$

where we use

$$\sum_{j=0}^{t-1}(1-\alpha)^{2j} \leq \sum_{j=0}^{\infty}(1-\alpha)^{2j} = \frac{1}{1-(1-\alpha)^2} = \frac{1}{\alpha(2-\alpha)} \leq \frac{1}{\alpha}, \forall\alpha\in(0,1).$$

Averaging the above inequality over i, we prove the first result in the lemma.

It follows

$$\begin{aligned}
\mathbb{E}\left[\left\|\mathbf{u}_{i,t} - g_i(\mathbf{w}_t)\right\|_2\right] &\leq \sqrt{\mathbb{E}\left[\left\|\mathbf{u}_{i,t} - g_i(\mathbf{w}_t)\right\|_2^2\right]} \\
&\leq \sqrt{\left(1-\frac{B\gamma}{4n}\right)^{2t}\left\|\mathbf{u}_{i,0} - g_i(\mathbf{w}_0)\right\|_2^2 + \frac{8n^2G_2^4G_1^2\eta^2}{B^2\gamma^2} + 4\gamma\sigma_0^2} \\
&\leq \left(1-\frac{B\gamma}{4n}\right)^{t}\left\|\mathbf{u}_{i,0} - g_i(\mathbf{w}_0)\right\|_2 + \frac{4nG_2^2G_1\eta}{B\gamma} + 2\gamma^{1/2}\sigma_0.
\end{aligned}$$

Averaging the above result, we prove the second result. □

Theorem 5.3 (Convergence of SONX-v1 with Lipschitz f_i) *Consider SONX-v1, and suppose Assumption 5.4 and 5.5 hold and f_i is G_1-Lipschitz continuous. Let $\eta_t = \eta = O(\frac{B\epsilon^6}{n\sigma_0^2})$, $\gamma_t = \gamma = O(\frac{\epsilon^4}{\sigma_0^2})$. Then after $T = O(\frac{n\sigma_0^2}{B\epsilon^8})$ iterations, we have $\mathbb{E}[\|\nabla F_{1/\bar{\rho}}(\mathbf{w}_t)\|_2^2] \leq O(\epsilon^2)$.*

Proof. From Lemma 5.10, we have

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\|\nabla F_{1/\bar{\rho}}(\mathbf{w}_t)\|_2^2\right] \leq \mathbb{E}\left[\frac{2\sum_{t=1}^{T}(F_{1/\bar{\rho}}(\mathbf{w}_t) - F_{1/\bar{\rho}}(\mathbf{w}_{t+1}))}{\eta T}\right] + \eta\bar{\rho}G^2$$
$$+ 4\bar{\rho}G_1\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\delta_t'\right] + 2\bar{\rho}\rho_1\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\delta_t\right].$$

Next, we bound the last two terms. From Lemma 5.11, we have

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\delta_t\right] \leq \frac{1}{T}\sum_{t=1}^{T}\left(1 - \frac{B\gamma}{4n}\right)^{2t}\delta_0 + \frac{8n^2G_2^4G_1^2\eta^2}{B^2\gamma^2} + 4\gamma\sigma_0^2.$$
$$\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\delta_t'\right] \leq \frac{1}{T}\sum_{t=1}^{T}\left(1 - \frac{B\gamma}{4n}\right)^{t}\delta_0' + \frac{4nG_2^2G_1\eta}{B\gamma} + 2\sqrt{\gamma}\sigma_0.$$

Since $\sum_{t=1}^{T}(1-\mu)^t \leq \frac{1}{\mu}$ for $\mu \in (0,1)$, we have

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\delta_t\right] \leq \frac{4n\delta_0}{B\gamma T} + \frac{8n^2G_2^4G_1^2\eta^2}{B^2\gamma^2} + 4\gamma\sigma_0^2.$$
$$\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\delta_t'\right] \leq \frac{4n\delta_0'}{B\gamma T} + \frac{4nG_2^2G_1\eta}{B\gamma} + 2\sqrt{\gamma}\sigma_0.$$

From Proposition 3.2, we have

$$\sum_{t=1}^{T}(F_{1/\bar{\rho}}(\mathbf{w}_t) - F_{1/\bar{\rho}}(\mathbf{w}_{t+1})) = F_{1/\bar{\rho}}(\mathbf{w}_1) - F_{1/\bar{\rho}}(\mathbf{w}_{T+1}) \leq F(\mathbf{w}_1) - F(\mathbf{w}_*).$$

Combining the above results, we have

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\|\nabla F_{1/\bar{\rho}}(\mathbf{w}_t)\|_2^2\right] \leq \mathbb{E}\left[\frac{2(F(\mathbf{w}_1) - F_*)}{\eta T}\right] + \eta\bar{\rho}G^2$$
$$+ 4\bar{\rho}G_1\left(\frac{4n\delta_0'}{B\gamma T} + \frac{4nG_2^2G_1\eta}{B\gamma} + 2\sqrt{\gamma}\sigma_0\right) + 2\bar{\rho}\rho_1\left(\frac{4n\delta_0}{B\gamma T} + \frac{8n^2G_2^4G_1^2\eta^2}{B^2\gamma^2} + 4\gamma\sigma_0^2\right).$$

Plugging the order of η, γ, we finish the proof.

□

Theorem 5.4 (Convergence of SONX-v1 with smooth f_i) *Consider SONX-v1, and suppose Assumption 5.1 and 5.5 hold and f_i is L_1-smooth. Let $\eta_t = \eta = O(\frac{B\epsilon^3}{n\sigma_0^2})$, $\gamma_t = \gamma = O(\frac{\epsilon^2}{\sigma_0^2})$, then after $T = O(\frac{n\sigma_0^2}{B\epsilon^5})$ iterations, we have $\mathbb{E}[\|\nabla F_{1/\bar{\rho}}(\mathbf{w}_t)\|_2^2] \leq O(\epsilon^2)$.*

Proof. By using the result for smooth f_i in Lemma 5.10, we have

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\|\nabla F_{1/\bar{\rho}}(\mathbf{w}_t)\|_2^2\right] \leq \mathbb{E}\left[\frac{2\sum_{t=1}^{T}(F_{1/\bar{\rho}}(\mathbf{w}_t) - F_{1/\bar{\rho}}(\mathbf{w}_{t+1}))}{\eta T}\right] + \eta\bar{\rho}G^2$$
$$+ \bar{\rho}(L_1 + 2\rho_1)\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\delta_t\right].$$

Plugging the bounds for the first and last term in the RHS, we have

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\|\nabla F_{1/\bar{\rho}}(\mathbf{w}_t)\|_2^2\right] \leq \mathbb{E}\left[\frac{2(F(\mathbf{w}_1) - F_*)}{\eta T}\right] + \eta\bar{\rho}G^2$$
$$+ \bar{\rho}(L_1 + 2\rho_1)\left(\frac{4n\delta_0}{B\gamma T} + \frac{8n^2G_2^4G_1^2\eta^2}{B^2\gamma^2} + 4\gamma\sigma_0^2\right).$$

Plugging the order of η, γ, we finish the proof.

□

Convergence of SONX-v2

Similar to the first option, we need to bound δ_t, δ_t' first.

Lemma 5.12 *Under Assumption 5.4, 5.5, by setting $\gamma_t = \gamma \leq \frac{1}{2}$, $\eta_t = \eta$, $\gamma_t' = \frac{n-B}{B(1-\gamma)} + (1-\gamma)$, we have:*

$$\mathbb{E}\left[\delta_t\right] \leq \left(1 - \frac{B\gamma}{2n}\right)^{2t}\delta_0 + 4\gamma\sigma_0^2 + \frac{24n^2G_2^4G_1^2\eta^2}{B^2\gamma},$$
$$\mathbb{E}\left[\delta_t'\right] \leq \left(1 - \frac{B\gamma}{2n}\right)^{t}\delta_0' + 2\gamma^{1/2}\sigma_0 + \frac{5nG_2^2G_1\eta}{B\gamma^{1/2}}.$$

Proof is omitted as it is similar to that of Lemma 5.11 but based on Lemma 5.5.

Theorem 5.5 (Convergence of SONX-v2) *Consider SONX-v2, and suppose Assumption 5.4, and 5.5 hold.*

- *If f_i is G_1-Lipschitz continuous, by setting $\eta = O(\frac{B\epsilon^4}{n\sigma_0})$, $\gamma = O(\frac{\epsilon^4}{\sigma_0^2})$, then after $T = O(\frac{n\sigma_0}{B\epsilon^6})$ iterations, we have $\mathbb{E}[\|\nabla F_{1/\bar{\rho}}(\mathbf{x}_t)\|_2^2] \leq \epsilon^2$.*

- *If f_i is further L_1-smooth, by setting $\eta = O(\frac{B\epsilon^2}{n\sigma_0})$, $\gamma = O(\frac{\epsilon^2}{\sigma_0^2})$, then the complexity reduces to $T = O(\frac{n\sigma_0}{B\epsilon^4})$.*

The proof follows similarly to that of Theorem 5.3 and Theorem 5.4 and is left as an exercise for interested readers.

5.3.2 SONEX for Non-smooth Outer functions

When f_i is Lipschitz continuous and non-smooth, the best complexity derived in last subsection is $O(n/(B\epsilon^6))$. Can we further improve the complexity when the inner functions are smooth? We present a method and its analysis in this section.

Let us make the following assumptions.

Assumption 5.6. *We assume that*

(i) $\mathbb{E}_{\zeta\sim\mathbb{P}_i}[\|g_i(\mathbf{w};\zeta) - g_i(\mathbf{w})\|_2^2] \leq \sigma_0^2$
(ii) $\mathbb{E}_{\zeta\sim\mathbb{P}_i}[\|\nabla g_i(\mathbf{w};\zeta) - \nabla g_i(\mathbf{w})\|_2^2] \leq \sigma_2^2$
(iii) $\mathbb{E}_{\zeta\sim\mathbb{P}_i}[\|\nabla g_i(\mathbf{w};\zeta)\|_2^2] \leq G_2^2$.

Assumption 5.7. *The following conditions hold:*

(i) f_i is ρ_1-weakly convex, G_1-Lipschitz continuous,
(ii) g_i is L_2-smooth and G_2-Lipschitz continuous.

Moreau Envelope Smoothing of the outer function

A classical approach of improving the convergence for non-smooth functions in convex optimization is smoothing, i.e., first smoothing the function and then using an optimizer for solving the resulting smoothed function. We define the Moreau envelope smoothing of f_i as follows:

$$\bar{f}_i(g) = \min_{\mathbf{u}\in\mathbb{R}^{d'}} \frac{\bar{\rho}_1}{2}\|\mathbf{u} - g\|_2^2 + f_i(\mathbf{u}), \tag{5.19}$$

where $\bar{\rho}_1 > \rho_1$. We present a lemma below regarding $\bar{f}_i$.

Lemma 5.13 *If f_i is G_1-Lipschitz continuous and ρ_1-weakly convex, then $\bar{f}_i$ is $\bar{L}_1$-smooth and G_1 Lipschitz continuous, where $\bar{L}_1 = \frac{\bar{\rho}_1(2\bar{\rho}_1-\rho_1)}{(\bar{\rho}_1-\rho_1)}$.*

Proof. Define $\text{prox}_{f_i/\bar{\rho}_1}(g) = \arg\min_{\mathbf{u}\in\mathbb{R}^{d'}} \frac{\bar{\rho}_1}{2}\|\mathbf{u} - g\|_2^2 + f_i(\mathbf{u})$. We have

$$\nabla\bar{f}_i(g) = \bar{\rho}_1(g - \text{prox}_{f_i/\bar{\rho}_1}(g)).$$

Due to the optimality condition of $\text{prox}_{f_i/\bar{\rho}_1}(g)$, we have

$$\bar{\rho}_1(g - \text{prox}_{f_i/\bar{\rho}_1}(g)) \in \partial f_i(\text{prox}_{f_i/\bar{\rho}_1}(g)).$$

Hence, $\nabla \bar{f}_i(g) \in \partial f_i(\text{prox}_{f_i/\bar{\rho}_1}(g))$, which implies $\|\nabla \bar{f}_i(g)\| \leq G_1$. The smoothness of $\bar{f}_i$ follows from Proposition 3.1. □

Relationship with Nesterov Smoothing

When f_i is a convex function, its Moreau envelope smoothing is also equivalent to the well-known Nesterov smoothing. To see this, let f_i^* denote the convex conjugate of f_i, i.e., $f_i^*(\mathbf{u}) = \max_{g \in \mathbb{R}^{d'}} \mathbf{u}^\top g - f_i(g)$. Since f_i is convex, we have $f_i(g) = \max_{\mathbf{u} \in \mathcal{U}} \mathbf{u}^\top g - f_i^*(\mathbf{u})$, where $\mathcal{U} = \text{dom}(f_i^*)$ is bounded as $\|\partial f_i(g)\| \leq G_1$. As a result,

$$\bar{f}_i(g) = \min_{\mathbf{u} \in \mathbb{R}^{d'}} \frac{\bar{\rho}_1}{2}\|\mathbf{u} - g\|_2^2 + f_i(\mathbf{u}) = \min_{\mathbf{u} \in \mathbb{R}^{d'}} \frac{\bar{\rho}_1}{2}\|\mathbf{u} - g\|_2^2 + \max_{\mathbf{u}' \in \mathcal{U}} \mathbf{u}'^\top \mathbf{u} - f_i^*(\mathbf{u}').$$

By Sion's minimax theorem, we can switch the min and max. Hence,

$$\bar{f}_i(g) = \max_{\mathbf{u}' \in \mathcal{U}} \min_{\mathbf{u} \in \mathbb{R}^{d'}} \frac{\bar{\rho}_1}{2}\|\mathbf{u} - g\|_2^2 + \mathbf{u}'^\top \mathbf{u} - f_i^*(\mathbf{u}').$$

By solving the minimization over $\mathbf{u}$ and plugging the optimal solution into the expression, we get

$$\bar{f}_i(g) = \max_{\mathbf{u}' \in \mathcal{U}} g^\top \mathbf{u}' - f_i^*(\mathbf{u}') - \frac{1}{2\bar{\rho}_1}\|\mathbf{u}'\|_2^2. \tag{5.20}$$

This is known as Nesterov smoothing of the function $f_i(g)$. When $\bar{\rho}_1$ is sufficiently large, we can prove that $\bar{f}_i$ is sufficiently close to f_i.

Example

Example 5.1. *Let us consider the Nesterov smoothing of the hinge function $f(x) = [x]_+$. Let $\bar{\rho}_1 = 1/\varepsilon$ for some small $\varepsilon \ll 1$. Then, the Nesterov smoothing of the hinge function is*

$$\bar{f}(x) = \max_{u \in [0,1]} ux - \frac{\varepsilon}{2}u^2 = \begin{cases} x - \frac{\varepsilon}{2} & \text{if } x \geq \varepsilon \\ \frac{x^2}{2\varepsilon} & \text{if } 0 < x < \varepsilon \\ 0 & \text{o.w.} \end{cases}.$$

This is also known as the smoothed hinge function.

Solving the smoothed problem

With a smoothed outer function $\bar{f}_i$, we consider optimizing the following problem with some proper value of $\bar{\rho}_1$:

$$\min_{\mathbf{w}} \bar{F}(\mathbf{w}) := \frac{1}{n}\sum_{i=1}^{n} \bar{f}_i(g_i(\mathbf{w})). \tag{5.21}$$

Following Lemma 4.3, $\bar{F}(\cdot)$ is smooth with a smoothness parameter $\bar{L}_F = G_1 L_2 + G_2^2 \bar{L}_1$.

The key concern is how the convergence of solving the above problem translates to the convergence of solving the original problem (5.1). To address this question, we introduce a new convergence measure, named approximate ϵ-stationarity.

Definition 5.1 (Approximate ϵ-stationary solution) A point $\mathbf{w}$ is an approximate ϵ-stationary solution to the original problem (5.1), if there exists $(\mathbf{u}_1, \ldots, \mathbf{u}_n)$ and $\lambda_i \in \partial f(\mathbf{u}_i), \forall i$ such that

$$\left\| \frac{1}{n}\sum_{i=1}^{n} \nabla g_i(\mathbf{w})\lambda_i \right\|_2 \leq \epsilon, \tag{5.22}$$

$$\|\mathbf{u}_i - g_i(\mathbf{w})\|_2 \leq O(\epsilon), \forall i. \tag{5.23}$$

We note that this condition is closely related to the KKT condition of the following equivalent constrained formulation of the original problem (5.1):

$$\min_{\mathbf{w},\mathbf{u}} \quad \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{u}_i) \tag{5.24}$$

$$\text{s.t.} \quad g_i(\mathbf{w}) = \mathbf{u}_i, \forall i. \tag{5.25}$$

The Lagrangian function of this constrained formulation is given by

$$F(\mathbf{w}, \mathbf{u}, \lambda) = \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{u}_i) + \sum_{i=1}^{n} \lambda_i^\top (g_i(\mathbf{w}) - \mathbf{u}_i).$$

A solution $(\mathbf{w}, \mathbf{u}, \lambda)$ satisfies the KKT condition, if

$$\frac{1}{n}\sum_{i=1}^{n} \nabla g_i(\mathbf{w})\lambda_i = 0, \quad \lambda_i \in \partial f_i(\mathbf{u}_i)$$

$$\mathbf{u}_i = g_i(\mathbf{w}).$$

Hence, an approximate ϵ-stationary solution satisfies the KKT condition approximately when $\epsilon \ll 1$.

If f_i is L_1-smooth, an approximate ϵ-stationary solution is also a standard $O(\epsilon)$-stationary solution. To see this, we have

Algorithm 17 SONEX

Input: learning rate schedules $\{\eta_t\}_{t=1}^T$, $\{\gamma_t\}_{t=1}^T$, $\{\beta_t\}_{t=1}^T$; starting points $\mathbf{w}_0$, $\mathbf{v}_0$, $\mathbf{u}_0$
$\mathbf{w}_1 = \mathbf{w}_0 - \eta_0 \mathbf{v}_0$
for $t = 1, \ldots, T$ **do**
  Draw a batch of samples $\mathcal{B}_t \subset [n]$
  for $i \in \mathcal{B}_t$ **do**
    Draw two samples $\zeta_{i,t} \sim \mathbb{P}_i$
    Update the inner function value estimators by

$$\text{v1:} \quad \mathbf{u}_{i,t} = (1-\gamma_t)\mathbf{u}_{i,t-1} + \gamma_t g_i(\mathbf{w}_t; \zeta_{i,t})$$
$$\text{v2:} \quad \mathbf{u}_{i,t} = (1-\gamma_t)\mathbf{u}_{i,t-1} + \gamma_t g_i\left(\mathbf{w}_t; \zeta_{i,t}\right) + \gamma_t' \left(g_i\left(\mathbf{w}_t; \zeta_{i,t}\right) - g_i\left(\mathbf{w}_{t-1}; \zeta_{i,t}\right)\right)$$

  end for
  Set $\mathbf{u}_{i,t} = \mathbf{u}_{i,t-1}, i \notin \mathcal{B}_t$
  Compute the vanilla gradient estimator $\mathbf{z}_t = \frac{1}{B}\sum_{i\in\mathcal{B}_t'} \nabla g_i(\mathbf{w}_t; \zeta_{i,t}')\nabla \bar{f}_i(\mathbf{u}_{i,t})$
  Update the MA gradient estimator $\mathbf{v}_t = (1-\beta_t)\mathbf{v}_{t-1} + \beta_t \mathbf{z}_t$
  Update the model $\mathbf{w}$ by $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{v}_t$
end for

$$\begin{aligned}
&\left\|\frac{1}{n}\sum_{i=1}^n \nabla g_i(\mathbf{w})\nabla f_i(g_i(\mathbf{w}))\right\|_2 \\
&= \left\|\frac{1}{n}\sum_{i=1}^n \nabla g_i(\mathbf{w})\nabla f_i(g_i(\mathbf{w})) - \frac{1}{n}\sum_{i=1}^n \nabla g_i(\mathbf{w})\nabla f_i(\mathbf{u}_i) + \frac{1}{n}\sum_{i=1}^n \nabla g_i(\mathbf{w})\nabla f_i(\mathbf{u}_i)\right\|_2 \\
&\leq \frac{1}{n}\sum_{i=1}^n G_2 L_1 \|\mathbf{u}_i - g_i(\mathbf{w})\|_2 + \epsilon \leq O(\epsilon).
\end{aligned}$$

The following proposition states that an ϵ-stationary solution to the smoothed problem (5.21) is an approximate ϵ-stationary solution to the original problem when $\bar{\rho}_1$ is sufficiently large.

Proposition 5.1 *Let* $\mathbf{w}$ *be an* ϵ*-stationary solution to (5.21), when* $\bar{\rho}_1 = 1/\epsilon$*, then* $\mathbf{w}$ *is also an approximate* ϵ*-stationary solution to (5.1).*

Proof. Given that $\mathbf{w}$ be an ϵ-stationary solution to (5.21), we have

$$\left\|\frac{1}{n}\sum_{i=1}^n \nabla g_i(\mathbf{w})\nabla \bar{f}_i(g_i(\mathbf{w}))\right\|_2 \leq \epsilon.$$

We define $\mathbf{u}_i = \text{prox}_{f_i/\bar{\rho}_1}(g_i(\mathbf{w})) = \arg\min_{\mathbf{u}} f_i(\mathbf{u}) + \frac{\bar{\rho}_1}{2}\|\mathbf{u} - g_i(\mathbf{w})\|_2^2$ and $\lambda_i = \nabla \bar{f}_i(g_i(\mathbf{w}))$. Since $\nabla \bar{f}_i(g_i(\mathbf{w})) \in \partial f_i(\text{prox}_{f_i/\bar{\rho}_1}(g_i(\mathbf{w}))) = \partial f_i(\mathbf{u}_i)$. As a result, we have $\lambda_i \in \partial f_i(\mathbf{u}_i)$ and $\left\|\frac{1}{n}\sum_{i=1}^n \nabla g_i(\mathbf{w})\lambda_i\right\|_2 \leq \epsilon$.

Due to the optimality condition of $\mathbf{u}_i$, we have $g_i(\mathbf{w}) - \mathbf{u}_i \in \partial f_i(\mathbf{u}_i)/\bar{\rho}_1$. Since f_i is G_1-Lipschitz continuous and $\bar{\rho}_1 \geq 1/\epsilon$, hence, $\|\mathbf{u}_i - g_i(\mathbf{w})\|_2 \leq O(\epsilon)$. □

Next, we discuss algorithms and complexities for solving the smoothed problem when $\bar{\rho}_1 = 1/\epsilon$. Since both inner and outer functions of the smoothed problem are smooth, we can leverage the moving average gradient estimators. We present detailed steps for solving the smoothed problem in Algorithm 17, which is referred to as SONEX.

A step in implementing SONEX for solving the smoothed problem (5.21) is the calculation of $\nabla \bar{f}_i(\mathbf{u}_{i,t})$, which amounts to solving a proximal mapping of f_i, i.e.,

$$\text{prox}_{f_i/\bar{\rho}_1}(\mathbf{u}_{i,t}) = \arg\min_{\mathbf{u}\in\mathbb{R}^{d'}} \frac{\bar{\rho}_1}{2}\|\mathbf{u} - \mathbf{u}_{i,t}\|_2^2 + f_i(\mathbf{u}).$$

In fact, $\nabla \bar{f}_i(\mathbf{u}_{i,t}) = \bar{\rho}_1(\mathbf{u}_{i,t} - \text{prox}_{f_i/\bar{\rho}_1}(\mathbf{u}_{i,t}))$.

Convergence of SONEX-v1

Finally, we present the complexity of SONEX-v1 for finding an ϵ-stationary solution to the smoothed problem when $\bar{\rho}_1 = 1/\epsilon$.

Corollary 5.1 (Convergence of SONEX-v1) *Under Assumptions 5.6 and 5.7, if we set* $\mathbf{u}_0$ *such that* $\frac{1}{n}\mathbb{E}[\sum_{i=1}^n \|\mathbf{u}_{i,0} - g_i(\mathbf{w}_0)\|_2^2] \leq O(\epsilon)$, $\beta = O(\frac{\epsilon^2}{\sigma^2})$, $\gamma = O(\frac{\epsilon^4}{\sigma_0^2})$, $\eta = \min(\epsilon, O(\beta\epsilon), O(\frac{B\epsilon\gamma}{n}))$, $\bar{\rho}_1 = 1/\epsilon > \rho_1$, *then SONEX-v1 finds an approximate* $O(\epsilon)$*-stationary solution to the original problem (5.1) with a complexity of* $O(\frac{n\sigma_0^2}{B\epsilon^7})$.

Proof. The proof can be completed by using the convergence result of SOX with noting the order of $\bar{L}_1 = O(\bar{\rho}_1) = O(1/\epsilon)$ and $L_F = O(\bar{L}_1) = O(1/\epsilon)$. □

Convergence of SONEX-v2

SONEX-v2 is a combination of SOX and MSVR, i.e., with $\mathbf{u}_t$ sequence from MSVR and $\mathbf{v}_t$ from SOX.

Theorem 5.6 (Convergence of SONEX-v2) *Under Assumptions 5.6 and 5.7, if we set* $\mathbf{u}_1$ *such that* $\frac{1}{n}\mathbb{E}[\sum_{i=1}^n \|\mathbf{u}_{i,0} - g_i(\mathbf{w}_0)\|_2^2] \leq O(\epsilon^3/\sigma_0)$, $\beta = O(\frac{\epsilon^2}{\sigma^2})$, $\gamma = O(\frac{\epsilon^2}{\sigma_0^2})$, $\eta = \min(O(\epsilon), O(\beta\epsilon), O(\frac{B\sqrt{\gamma}\epsilon}{n}))$ *and* $\bar{\rho}_1 = \frac{1}{\epsilon} > \rho_1$, *then SONEX-v2 finds an approximate* ϵ*-stationary solution to the original problem (5.1) with a complexity of*

$$T = O\left(\max\left\{\frac{1}{\epsilon^3}, \frac{\sigma^2}{\epsilon^5}, \frac{n\sigma_0}{B\epsilon^5}\right\}\right),$$

where $\sigma^2 = \frac{G_1^2\sigma_2^2}{B} + \frac{G_1^2 G_2^2(n-B)}{B(n-1)}$.

Proof. The proof is similar to that of Theorem 4.3 except that the $\diamond$ inequality in Lemma 4.10 is replaced by the following for using MSVR estimators (see Lemma 5.5):

$$(\diamond) \quad \mathbb{E}\left[\delta_{t+1}\right] \leq \mathbb{E}[(1-\bar{\gamma})\delta_t + C_3\eta^2\Gamma_t + \bar{\gamma}^2\sigma'^2],$$

where $\bar{\gamma}_{=}\frac{B\gamma}{n}$, $\sigma'^2 = \frac{2n\sigma_0^2}{B}$, $C_3 = O(n/B)$

We only highlight the changes below and leave details as an exercise. First, the condition on η in Lemma 4.10 is changed to

$$\eta \leq O\left(\frac{1}{L}, \frac{\beta}{\sqrt{C_2}}, \sqrt{\frac{\bar{\gamma}}{C_1 C_3}}\right).$$

The settings on $\beta, \bar{\gamma}$ remain the same as $\beta = O(\frac{\epsilon^2}{\sigma^2})$, $\bar{\gamma} = O(\frac{\epsilon^2}{C_1\sigma'^2})$. The iteration complexity becomes:

$$\begin{aligned} T &= O\left(\max\left\{\frac{C_\Upsilon L_F}{\epsilon^2}, \frac{C_\Upsilon\sqrt{C_2}}{\epsilon^2\beta}, \frac{C_\Upsilon\sqrt{C_1 C_3}}{\sqrt{\bar{\gamma}}\epsilon^2}\right\}\right) \\ &= O\left(\max\left\{\frac{C_\Upsilon L_F}{\epsilon^2}, \frac{C_\Upsilon\sigma^2\sqrt{C_2}}{\epsilon^4}, \frac{C_\Upsilon\sqrt{C_3}C_1\sigma'}{\epsilon^3}\right\}\right). \end{aligned}$$

and C_Υ is changed to

$$\begin{aligned} C_\Upsilon &= A_0 - A_* + \frac{\eta}{\beta}\Delta_0 + \frac{C_1\eta}{\bar{\gamma}}\delta_0 \leq A_0 - A_* + O\left(\frac{1}{\sqrt{C_2}}\right)\Delta_0 + O\left(\frac{\sqrt{C_1}}{\sqrt{C_3\bar{\gamma}}}\right)\delta_0 \\ &= A_0 - A_* + O\left(\frac{1}{\sqrt{C_2}}\right)\Delta_0 + O\left(\frac{C_1\sigma'}{\sqrt{8C_3}\epsilon}\right)\delta_0. \end{aligned}$$

Then, as in the proof of Theorem 5.1, we substitute $C_1 = O(\bar{L}_1^2)$, $C_2 = O(\bar{L}_F^2)$, $C_3 = O(n/B)$, $\sigma^2 = \frac{G_1^2\sigma_2^2}{B} + \frac{G_1^2G_2^2(n-B)}{B(n-1)}$, and $\sigma'^2 = O(n\sigma_0^2/B)$ into the above complexity expression and C_Υ, and obtain

$$\begin{aligned} T &= O\left(\max\left\{\frac{C_\Upsilon\bar{L}_F}{\epsilon^2}, \frac{C_\Upsilon\bar{L}_F\sigma^2}{\epsilon^4}, \frac{C_\Upsilon n\bar{L}_1^2\sigma_0}{B\epsilon^3}\right\}\right), \\ C_\Upsilon &\leq O(F(\mathbf{w}_0) - \bar{F}_*) + O\left(\frac{1}{\bar{L}_F}\right)\Delta_0 + O\left(\frac{\bar{L}_1^2\sigma_0}{\epsilon}\right)\delta_0. \end{aligned}$$

We finish the proof by noting that $\bar{L}_1 = O(1/\epsilon)$ and $\bar{L}_F = O(1/\epsilon)$ and $C_\Upsilon = O(1)$ if $\delta_0 \leq O(\epsilon^3/\sigma_0)$.

□

5.4 Convex inner and outer functions

In Chapter 3, we discussed standard stochastic convex optimization and established the iteration complexities of various algorithms. For general convex problems,

Algorithm 18 ALEXR

Input: learning rate schedules $\{\eta_t\}_{t=1}^T$, $\{\alpha_t\}_{t=1}^T$, $\theta \in [0,1]$; starting points $\mathbf{w}_0$, $\mathbf{y}_1 \in \mathcal{Y}_1 \times \cdots \times \mathcal{Y}_n$
Let $\mathbf{w}_1 = \mathbf{w}_0$
for $t = 1, \ldots, T-1$ **do**
  Sample a batch $\mathcal{B}_t \subset \{1, \ldots, n\}$, $|\mathcal{B}_t| = B$
  for each $i \in \mathcal{S}_t$ **do**
    Draw a sample $\zeta_{i,t}, \zeta'_{i,t} \sim \mathbb{P}_i$
    Compute $\tilde{g}_{i,t} = g_i(\mathbf{w}_t; \zeta_{i,t}) + \theta(g_i(\mathbf{w}_t; \zeta_{i,t}) - g_i(\mathbf{w}_{t-1}; \zeta_{i,t}))$
    Update $y_{i,t+1} = \arg\max_{y_i \in \mathcal{Y}_i} \left\{ y_i^\top \tilde{g}_{i,t} - f_i^*(y_i) - \frac{1}{\alpha_t} D_{\psi_i}(y_i, y_{i,t}) \right\}$
  end for
  For each $i \notin \mathcal{B}_t$, $y_{i,t+1} = y_{i,t}$
  Compute the vanilla gradient estimator $\mathbf{z}_t = \frac{1}{B} \sum_{i \in \mathcal{B}_t} [\partial g_i(\mathbf{w}_t; \zeta'_{i,t})]^\top y_{i,t+1}$
  Update $\mathbf{w}_{t+1} = \arg\min_{\mathbf{w}} \left\{ \mathbf{z}_t^\top \mathbf{w} + \frac{1}{2\eta_t} \|\mathbf{w} - \mathbf{w}_t\|_2^2 + r(\mathbf{w}) \right\}$
end for

stochastic gradient descent (SGD) achieves a complexity of $O(1/\epsilon^2)$, while for μ-strongly convex problems, its complexity improves to $O(1/(\mu\epsilon))$. These analyses rely on the assumption of unbiased stochastic gradient estimators, which does not hold for convex compositional optimization problems.

In this section, we introduce stochastic algorithms for a family of convex FCCO problems, where both the inner and outer functions are convex. We establish that these algorithms attain the same order of iteration complexities as SGD in standard stochastic convex optimization. In particular, let us consider a regularized convex FCCO:

$$\min_{\mathbf{w} \in \mathbb{R}^d} F(\mathbf{w}) := \frac{1}{n} \sum_{i=1}^n f_i(g_i(\mathbf{w})) + r(\mathbf{w}), \tag{5.26}$$

where $g_i(\mathbf{w}) = \mathbb{E}_{\zeta \sim \mathbb{P}_i}[g_i(\mathbf{w}; \zeta)]$, the outer and inner functions satisfy the following assumption.

Assumption 5.8. *The following conditions hold:*

(i) f_i is convex, G_1-Lipschitz continuous, and $\partial f_i(\cdot) \geq 0$.
(ii) g_i is convex and G_2-Lipschitz continuous.
(iii) r is μ-strongly convex for some $\mu \geq 0$.

Group DRO (5.2) could satisfy the above assumption when the individual loss function is convex and Lipschitz with respect to the model parameter. Two-way partial AUC maximization considered in Section 6.4.3 is another example satisfying the above assumption when the loss function is convex and Lipschitz continuous.

Let f_i^* denote the convex conjugate of f_i. We can write $f_i(g_i(\mathbf{w}))$ as

$$f_i(g_i(\mathbf{w})) = \max_{y_i \in \mathcal{Y}_i} (y_i^\top g_i(\mathbf{w}) - f_i^*(y_i)),$$

where $\mathcal{Y}_i = \mathrm{dom}(f_1^*)$. Since $0 \leq \partial f_i(\cdot)$ and $\|\partial f_i(\cdot)\| \leq G_1$, hence $\mathcal{Y}_i$ is a compact set following from Lemma 1.8.

Then, we can convert (5.26) into an equivalent minimax optimization problem:

$$\min_{\mathbf{w}\in\mathbb{R}^d} \max_{\mathbf{y}\in\mathcal{Y}} \frac{1}{n}\sum_{i=1}^{n}(y_i^\top g_i(\mathbf{w}) - f_i^*(y_i)) + r(\mathbf{w}), \tag{5.27}$$

where $\mathbf{y} = (y_1, \ldots, y_n)^\top$, $\mathcal{Y} = \mathcal{Y}_1 \times \cdots \mathcal{Y}_n$. Thus, the above problem is convex-concave problem under Assumption 5.8.

We introduce a method to optimize the above minimax problem. However, there are several unique challenges: (i) updating all coordinates of $\mathbf{y}$ is difficult because it is computationally prohibitive to traverse all data points $i = 1, \ldots, n$ at each iteration; (ii) we only have access to stochastic evaluations of the functions $g_i(\mathbf{w}; \zeta)$, which limits our ability to update both the corresponding coordinate of $\mathbf{y}$ and the parameter $\mathbf{w}$.

5.4.1 The ALEXR Algorithm

To present the algorithm, we assume a strongly convex prox-function ψ_i for the i-th coordinate and impose the following conditions.

Assumption 5.9. *Suppose ψ_i is differentiable and obeys the following conditions*

(i) ψ_i is μ_ψ-strongly convex with respect to $\|\cdot\|_2$, i.e., $\psi_i(y) \geq \psi_i(y') + \nabla\psi_i(y')^\top(y - y') + \frac{\mu_\psi}{2}\|y - y'\|_2^2$.
(ii) $D_{f_i^*}(y, y') \geq \rho D_{\psi_i}(y, y')$ for some $\rho \geq 0$.
(iii) The following proximal mapping can be easily computed:

$$y_{i,t+1} = \arg\max_{y_i\in\mathcal{Y}_i} \left\{ y_i^\top \tilde{g}_{i,t} - f_i^*(y_i) - \frac{1}{\alpha_t} D_{\psi_i}(y_i, y_{i,t}) \right\}.$$

A meta-algorithm, termed ALEXR, is presented in Algorithm 18. ALEXR employs stochastic block-coordinate proximal mirror ascent to update $\mathbf{y}$, using the prox-function ψ_i for the i-th coordinate, and applies stochastic proximal gradient descent to update $\mathbf{w}$. Below, we consider different choices of the prox-functions ψ_i and the corresponding updates for $y_{i,t+1}$.

ALEXR-v1 for smooth f_i: using $\psi_i = f_i^*$

When f_i is L_1-smooth, its convex conjugate f_i^* is $1/L_1$-strongly convex. We can use $\psi_i = f_i^*$ to define a Bregman divergence $D_{\psi_i}(y, y') = D_{f_i^*}(y, y')$.

Critical: In this case, Assumption 5.9 (i) and (ii) hold with $\mu_\psi = 1/L_1$, and $\rho = 1$.

Let us consider the update of $y_{i,t+1}$, which becomes:

$$y_{i,t+1} = \arg\max_{y_i \in \mathcal{Y}_i} \left\{ y_i^\top \tilde{g}_{i,t} - f_i^*(y_i) - \frac{1}{\alpha_t} D_{f_i^*}(y_i, y_{i,t}) \right\}, \forall i \in \mathcal{B}_t. \tag{5.28}$$

The following lemma provides an efficient way to compute $y_{i,t+1}$, which also builds the connection to the sequence of $\mathbf{u}_{i,t}$ in SOX and MSVR.

Lemma 5.14 *Let $\mathbf{u}_{i,t-1} \in \partial f_i^*(y_{i,t})$. Then for $i \in \mathcal{B}_t$ we have $y_{i,t+1} = \nabla f_i(\mathbf{u}_{i,t})$, where $\mathbf{u}_{i,t} = \frac{1}{1+\alpha_t}\mathbf{u}_{i,t-1} + \frac{\alpha_t}{1+\alpha_t}\tilde{g}_{i,t}$.*

Proof. For the problem (5.28), we have

$$\begin{aligned}
&y_i^\top \tilde{g}_{i,t} - f_i^*(y_i) - \frac{1}{\alpha_t} D_{f_i^*}(y_i, y_{i,t}) \\
&= y_i^\top \tilde{g}_{i,t} - f_i^*(y_i) - \frac{1}{\alpha_t}(f_i^*(y_i) - \partial f_i^*(y_{i,t})^\top (y_i - y_{i,t}) - f_i^*(y_{i,t})) \\
&= y_i^\top (\tilde{g}_{i,t} + \frac{1}{\alpha_t}\partial f_i^*(y_{i,t})) - (1 + \frac{1}{\alpha_t}) f_i^*(y_i) - \frac{1}{\alpha_t}\partial f_i^*(y_{i,t})^\top y_{i,t} + \frac{1}{\alpha_t} f_i^*(y_{i,t}).
\end{aligned}$$

Hence $y_{i,t+1} \in \arg\max_{y_i \in \mathcal{Y}_i} y_i^\top (\frac{\alpha_t}{1+\alpha_t}\tilde{g}_{i,t} + \frac{1}{1+\alpha_t}\partial f_i^*(y_{i,t})) - f_i^*(y_i)$. If we define $\mathbf{u}_{i,t} = \frac{\alpha_t}{1+\alpha_t}\tilde{g}_{i,t} + \frac{1}{1+\alpha_t}\partial f_i^*(y_{i,t})$, we have

$$f(\mathbf{u}_{i,t}) = \max_{y_i \in \mathcal{Y}_i} y_i^\top \mathbf{u}_{i,t} - f_i^*(y_i) = y_{i,t+1}^\top \mathbf{u}_{i,t} - f_i^*(y_{i,t+1}).$$

Hence, $\mathbf{u}_{i,t} \in \arg\max_{\mathbf{u}} y_{i,t+1}^\top \mathbf{u} - f_i(\mathbf{u})$ and therefore $y_{i,t+1} = \nabla f_i(\mathbf{u}_{i,t})$. □

If f_i is a Legendre function such that $\nabla f_i^{-1} = \nabla f_i^*$ (see Lemma 1.8). Then, we can derive the following equivalent update of $\mathbf{u}$ sequence such that $y_{i,t} = \nabla f_i(\mathbf{u}_{i,t-1})$.

$$\mathbf{u}_{i,t} = \begin{cases} \frac{1}{1+\alpha_t}\mathbf{u}_{i,t-1} + \frac{\alpha_t}{1+\alpha_t}\tilde{g}_{i,t}, & \text{if } i \in \mathcal{B}_t \\ \mathbf{u}_{i,t-1} & \text{o.w.} \end{cases}. \tag{5.29}$$

When $\theta = 0$, the equivalent $\mathbf{u}$ update (5.64) becomes:

$$\mathbf{u}_{i,t} = (1 - \frac{\alpha_t}{1+\alpha_t})\mathbf{u}_{i,t-1} + \frac{\alpha_t}{1+\alpha_t} g_i(\mathbf{w}_t; \zeta_{i,t}), \forall i \in \mathcal{B}_t. \tag{5.30}$$

This is the same as the moving average estimator in SOX with $\gamma_t = \alpha_t/(1+\alpha_t)$. Using the equivalent $\mathbf{u}$ sequence, the stochastic gradient estimator becomes $\mathbf{z}_t = \frac{1}{B}\sum_{i \in \mathcal{B}_t} [\partial g_i(x_t; \zeta'_{i,t})]^\top \nabla f_i(\mathbf{u}_{i,t})$. If the regularizer r is not present, the update of the model parameter becomes $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{z}_t$. In this case, ALEXR with $\theta = 0$ is the same as SOX with $\beta_t = 1$. We will prove its convergence for convex and strongly convex regularizer r.

When $\theta > 0$, the equivalent $\mathbf{u}$ update (5.64) becomes:

$$\mathbf{u}_{i,t} = (1 - \frac{\alpha_t}{1+\alpha_t})\mathbf{u}_{i,t-1} + \frac{\alpha_t}{1+\alpha_t} g_i(\mathbf{w}_t; \zeta_{i,t}) + \frac{\theta\alpha_t}{1+\alpha_t}(g_i(\mathbf{w}_t; \zeta_t) - g_i(\mathbf{w}_{t-1}; \zeta_t)). \tag{5.31}$$

This is similar to the MSVR estimator with $\gamma_t = \frac{\alpha_t}{1+\alpha_t}$ and $\gamma_t' = \frac{\theta\alpha_t}{1+\alpha_t}$. However, the key difference is that γ_t' in MSVR is larger than 1, while it is smaller than 1 in ALEXR for convex problems. In practice, setting $\gamma_t' < 1$ is a better choice. We will prove a better convergence of ALEXR with $\theta \in (0, 1)$ for a strongly convex r.

ALEXR-v2 for non-smooth f_i: using a quadratic function $\psi_i(\cdot)$

When f_i is non-smooth, we cannot use f_i^* as the prox function. In this case, we will use a smooth and strongly convex ψ_i. a quadratic function $\psi_i(y) = \frac{1}{2}\|y\|_2^2$.

Critical: In this case, Assumption 5.9 (i) holds with $\mu_\psi = 1$, and Assumption 5.9 (ii) holds with $\rho = 0$.

Example

Example 5.2. *For the update of $y_{i,t+1}$, consider the example $f_i(\cdot) = [\,\cdot\,]_+$, as used in GDRO and TPAUC maximization. In this case, the conjugate $f_i^*(y)$ is the indicator function of the interval $[0, 1]$. Consequently, $y_{i,t+1}$ can be computed as:*

$$y_{i,t+1} = \arg\max_{y_i \in [0,1]} \left\{ y_i^\top \tilde{g}_{i,t} - \frac{1}{2\alpha_t}(y_i - y_{i,t})^2 \right\} = \Pi_{[0,1]}(y_{i,t} - \alpha_t \tilde{g}_{i,t}), \forall i \in \mathcal{B}_t,$$

where $\Pi_{[0,1]}(\cdot)$ projects the input into the range of $[0, 1]$.

5.4.2 Technical Lemmas

To facilitate the analysis, we define $(\mathbf{w}_*, \mathbf{y}_*)$ as the saddle point to the minimax problem and

$$
\begin{aligned}
&F(\mathbf{w},\mathbf{y}) = \frac{1}{n}\sum_{i=1}^{n} y_i^\top g_i(\mathbf{w}) - f_i^*(y_i) + r(\mathbf{w}),\\
&\tilde{g}_t = (\tilde{g}_{1,t},\dots,\tilde{g}_{n,t})^\top,\\
&\bar{y}_{i,t+1} = \arg\max_{y_i\in\mathcal{Y}_i}\left\{y_i^\top \tilde{g}_{i,t} - f_i^*(y_i) - \frac{1}{\alpha_t}D_{\psi_i}(y_i,y_{i,t})\right\}, \forall i\in[n]\\
&D_\psi(\mathbf{y},\mathbf{y}') = \sum_{i=1}^{n} D_{\psi_i}(y_i,y_i').
\end{aligned}
$$

Note that $\bar{\mathbf{y}}_{t+1}$ is a virtual sequence, which is updated for all coordinates from $\mathbf{y}_t$ making it independent of $\mathcal{B}_t$.

We make the following assumption regarding the stochastic estimators.

Assumption 5.10. *We assume that*

(i) $\mathbb{E}_{\zeta\sim\mathbb{P}_i}[\|g_i(\mathbf{w};\zeta) - g_i(\mathbf{w})\|_2^2] \le \sigma_0^2$.

(ii) $\mathbb{E}_{\zeta\sim\mathbb{P}_i}[\|\nabla g_i(\mathbf{w};\zeta) - \nabla g_i(\mathbf{w})\|_2^2] \le \sigma_2^2$.

(iii) $\mathbb{E}_{i\sim\mathbb{U}_n}\left[\left\|y_i\nabla g_i(\mathbf{w}) - \frac{1}{n}\sum_{i=1}^n y_i\nabla g_i(\mathbf{w})\right\|_2^2\right] \le \delta^2$ *for any fixed* $\mathbf{y}$*, where* $\mathbb{U}_n$ *denotes a uniform distribution.*

Lemma 5.15 *The following holds for any* $\mathbf{w},\mathbf{y}\in\mathcal{Y}$ *after the* t*-th iteration of Algorithm 18.*

$$
\begin{aligned}
&F(\mathbf{w}_{t+1},\mathbf{y}) - F(\mathbf{w},\bar{\mathbf{y}}_{t+1}) \qquad (5.32)\\
&\le \frac{1}{2\eta_t}\|\mathbf{w}-\mathbf{w}_t\|_2^2 - (\frac{1}{2\eta_t}+\frac{\mu}{2})\|\mathbf{w}-\mathbf{w}_{t+1}\|_2^2 - \frac{1}{2\eta_t}\|\mathbf{w}_{t+1}-\mathbf{w}_t\|_2^2\\
&+ A_t(\mathbf{y}) + B_t(\mathbf{y}) + C_t(\mathbf{w}),
\end{aligned}
$$

where

$$
\begin{aligned}
A_t(\mathbf{y}) &= \frac{1}{n\alpha_t}D_\psi(\mathbf{y},\mathbf{y}_t) - (\frac{1}{n\alpha_t}+\frac{\rho}{n})D_\psi(\mathbf{y},\bar{\mathbf{y}}_{t+1}) - \frac{1}{n\alpha_t}D_\psi(\bar{\mathbf{y}}_{t+1},\mathbf{y}_t)\\
B_t(\mathbf{y}) &= \frac{1}{n}\sum_{i=1}^{n}(g_i(\mathbf{w}_{t+1}) - \tilde{g}_{i,t})^\top(y_i - \bar{y}_{i,t+1})\\
C_t(\mathbf{w}) &= \frac{1}{n}\sum_{i=1}^{n}(g_i(\mathbf{w}_{t+1}) - g_i(\mathbf{w}))^\top\bar{y}_{i,t+1} - \mathbf{z}_t^\top(\mathbf{w}_{t+1}-\mathbf{w}).
\end{aligned}
$$

Proof. Following Lemma 3.10, for all $i\in[n]$ the dual update rule implies that for any $y\in\mathcal{Y}$ it holds

$$
\begin{aligned}
&\tilde{g}_{i,t}^\top(y_i - \bar{y}_{i,t+1}) + f_i^*(\bar{y}_{i,t+1}) - f_i^*(y_i)\\
&\le \frac{1}{\alpha_t}D_{\psi_i}(y_i,y_{i,t}) - (\frac{1}{\alpha_t}+\rho)D_{\psi_i}(y_i,\bar{y}_{i,t+1}) - \frac{1}{\alpha_t}D_{\psi_i}(\bar{y}_{i,t+1},y_{i,t}).
\end{aligned}
$$

Averaging this inequality over $i = 1,\dots,n$.

$$\frac{1}{n}\sum_{i=1}^{n}\tilde{g}_{i,t}^{\top}(y_{i,t}-\bar{y}_{i,t+1})+\frac{1}{n}\sum_{i=1}^{n}f_i^*(\bar{y}_{i,t+1})-\frac{1}{n}\sum_{i=1}^{n}f_i^*(y_i) \tag{5.33}$$
$$\leq \frac{1}{n\alpha_t}D_\psi(\mathbf{y},\mathbf{y}_t)-(\frac{1}{n\alpha_t}+\frac{\rho}{n})D_\psi(\mathbf{y},\bar{\mathbf{y}}_{t+1})-\frac{1}{n\alpha_t}D_\psi(\bar{\mathbf{y}}_{t+1},\mathbf{y}_t).$$

According to Lemma 3.6, the primal update rule implies that

$$\mathbf{z}_t^{\top}(\mathbf{w}_{t+1}-\mathbf{w})+r(\mathbf{w}_{t+1})-r(\mathbf{w}) \tag{5.34}$$
$$\leq \frac{1}{2\eta_t}\|\mathbf{w}-\mathbf{w}_t\|_2^2-(\frac{1}{2\eta_t}+\frac{\mu}{2})\|\mathbf{w}-\mathbf{w}_{t+1}\|_2^2-\frac{1}{2\eta_t}\|\mathbf{w}_{t+1}-\mathbf{w}_t\|_2^2.$$

By the definition of $F(\mathbf{w},\mathbf{y})$, we have

$$\begin{aligned}
&F(\mathbf{w}_{t+1},\mathbf{y})-F(\mathbf{w},\bar{\mathbf{y}}_{t+1})\\
&=\frac{1}{n}\sum_{i=1}^{n}y_i^{\top}g_i(\mathbf{w}_{t+1})-\frac{1}{n}\sum_{i=1}^{n}f_i^*(y_i)+r(\mathbf{w}_{t+1})-\frac{1}{n}\sum_{i=1}^{n}\bar{y}_{i,t+1}^{\top}g_i(\mathbf{w})\\
&+\frac{1}{n}\sum_{i=1}^{n}f_i^*(\bar{y}_{i,t+1})-r(\mathbf{w})\\
&=\frac{1}{n}\sum_{i=1}^{n}g_i(\mathbf{w}_{t+1})^{\top}(y_i-\bar{y}_{i,t+1})+\frac{1}{n}\sum_{i=1}^{n}f_i^*(\bar{y}_{i,t+1})-\frac{1}{n}\sum_{i=1}^{n}f_i^*(y_i)\\
&+\frac{1}{n}\sum_{i=1}^{n}(g_i(\mathbf{w}_{t+1})-g_i(\mathbf{w}))^{\top}\bar{y}_{i,t+1}+r(\mathbf{w}_{t+1})-r(\mathbf{w}).
\end{aligned}$$

Combining the equation above with (5.34) and (5.33), we can finish the proof. □

Next, we bound the three terms $A_t(\mathbf{y}), B_t(\mathbf{y}), C_t(\mathbf{w})$ separately.

Lemma 5.16 *Let $\tau_t = 1/\alpha_t$. For $\mathbf{y}$ that possibly depends on all randomness in the algorithm and any $\lambda_0 > 0$, we have*

$$\mathbb{E}[A_t(\mathbf{y})]=\mathbb{E}\left[\frac{\tau_t}{n}D_\psi(\mathbf{y},\mathbf{y}_t)-\frac{\tau_t+\rho}{n}D_\psi(\mathbf{y},\bar{\mathbf{y}}_{t+1})-\frac{\tau_t}{n}D_\psi(\bar{\mathbf{y}}_{t+1},\mathbf{y}_t)\right] \tag{5.35}$$
$$\leq \mathbb{E}\left[\frac{\tau_t+\rho\left(1-\frac{B}{n}\right)}{B}D_\psi(\mathbf{y},\mathbf{y}_t)-\frac{\tau_t+\rho}{B}D_\psi(\mathbf{y},\mathbf{y}_{t+1})\right]-\frac{\tau_t}{n}\mathbb{E}\left[D_\psi(\bar{\mathbf{y}}_{t+1},\mathbf{y}_t)\right]$$
$$+\mathbb{E}\left[\frac{\lambda_0(\tau_t+\rho)}{n}(D_\psi(\mathbf{y},\hat{\mathbf{y}}_t)-D_\psi(\mathbf{y},\hat{\mathbf{y}}_{t+1}))\right]$$
$$+\frac{(n-B)(\tau_t+\rho)}{2\mu_\psi\lambda_0 nB}\mathbb{E}\left[\sum_{i=1}^{n}\left\|\nabla\psi_i(\bar{y}_{i,t+1})-\nabla\psi_i(y_{i,t})\right\|_2^2\right],$$

where the sequence $\{\hat{\mathbf{y}}_t\}_t, \hat{\mathbf{y}}_t \in \mathcal{Y}$ is virtual. In addition, for $\mathbf{y}_*$, we have

$$\mathbb{E}[A_t(\mathbf{y}_*)] = \mathbb{E}\left[\frac{\tau_t}{n}D_\psi(\mathbf{y}_*, \mathbf{y}_t) - \frac{\tau_t+\rho}{n}D_\psi(\mathbf{y}_*, \bar{\mathbf{y}}_{t+1}) - \frac{\tau_t}{n}D_\psi(\bar{\mathbf{y}}_{t+1}, \mathbf{y}_t)\right] \tag{5.36}$$
$$\leq \mathbb{E}\left[\frac{\tau_t+\rho\left(1-\frac{B}{n}\right)}{B}D_\psi(\mathbf{y}_*, \mathbf{y}_t) - \frac{\tau_t+\rho}{B}D_\psi(\mathbf{y}_*, \mathbf{y}_{t+1})\right] - \frac{\tau_t}{n}\mathbb{E}\left[D_\psi(\bar{\mathbf{y}}_{t+1}, \mathbf{y}_t)\right].$$

Proof.

$$\begin{aligned}
&\frac{\tau_t}{n}D_\psi(\mathbf{y}, \mathbf{y}_t) - \frac{\tau_t+\rho}{n}D_\psi(\mathbf{y}, \bar{\mathbf{y}}_{t+1}) - \frac{\tau_t}{n}D_\psi(\bar{\mathbf{y}}_{t+1}, \mathbf{y}_t) \qquad (5.37)\\
&= \frac{\tau_t+\rho\left(1-\frac{B}{n}\right)}{B}D_\psi(\mathbf{y}, \mathbf{y}_t) - \frac{\tau_t+\rho}{B}D_\psi(\mathbf{y}, \mathbf{y}_{t+1}) - \frac{\tau_t}{n}D_\psi(\bar{\mathbf{y}}_{t+1}, \mathbf{y}_t)\\
&+ \left(\frac{\tau_t+\rho}{B}D_\psi(\mathbf{y}, \mathbf{y}_{t+1}) - \frac{\tau_t+\rho}{n}D_\psi(\mathbf{y}, \bar{\mathbf{y}}_{t+1}) + \frac{(B-n)(\tau_t+\rho)}{nB}D_\psi(\mathbf{y}, \mathbf{y}_t)\right).
\end{aligned}$$

For bounding the last three terms, we consider the following:

$$\begin{aligned}
&\frac{1}{B}D_{\psi_i}(y_i, y_{i,t+1}) - \frac{1}{n}D_{\psi_i}(y_i, \bar{y}_{i,t+1}) + \frac{(B-n)}{nB}D_{\psi_i}(y_i, y_{i,t}) \qquad (5.38)\\
&= \frac{1}{B}\left(\psi_i(y_i) - \psi_i(y_{i,t+1}) - \nabla\psi_i(y_{i,t+1})^\top(y_i - y_{i,t+1})\right)\\
&- \frac{1}{n}\left(\psi_i(y_i) - \psi_i(\bar{y}_{i,t+1}) - \nabla\psi_i(\bar{y}_{i,t+1})^\top(y_i - \bar{y}_{i,t+1})\right)\\
&+ \frac{(B-n)}{nB}\left(\psi_i(y_i) - \psi_i(y_{i,t}) - \nabla\psi_i(y_{i,t})^\top(y_i - y_{i,t})\right)\\
&= \left[\frac{1}{n}\left(\psi_i(\bar{y}_{i,t+1}) - \frac{n}{B}\psi_i(y_{i,t+1}) + \frac{n-B}{B}\psi_i(y_{i,t})\right)\right]\\
&+ \left[\frac{1}{B}\nabla\psi_i(y_{i,t+1})^\top y_{i,t+1} - \frac{1}{n}\nabla\psi_i(\bar{y}_{i,t+1})^\top\bar{y}_{i,t+1} + \frac{(B-n)}{nB}\nabla\psi_i(y_{i,t})^\top y_{i,t}\right]\\
&+ \underbrace{\frac{1}{n}(-\frac{n}{B}\nabla\psi_i(y_{i,t+1}) + \nabla\psi_i(\bar{y}_{i,t+1}) + \frac{n-B}{B}\nabla\psi_i(y_{i,t}))^\top y_i}_{\sharp}.
\end{aligned}$$

Taking expectation over $\mathcal{B}_t$ for the first two terms in the brackets of the above bound will give zeros. This is because that both $\bar{y}_{i,t+1}$ and $y_{i,t}$ are independent of $\mathcal{B}_t$ such that

$$\begin{aligned}
\mathbb{E}_{\mathcal{B}_t}[\psi_i(y_{i,t+1})] &= \frac{B}{n}\psi_i(\bar{y}_{i,t+1}) + \frac{n-B}{n}\psi_i(y_{i,t}),\\
\mathbb{E}_{\mathcal{B}_t}\left[\nabla\psi_i(y_{i,t+1})^\top y_{i,t+1}\right] &= \frac{B}{n}\nabla\psi_i(\bar{y}_{i,t+1})^\top\bar{y}_{i,t+1} + \frac{n-B}{n}\nabla\psi_i(y_{i,t})^\top y_{i,t},\\
\mathbb{E}_{\mathcal{B}_t}\left[\nabla\psi_i(y_{i,t+1})\right] &= \frac{B}{n}\nabla\psi_i(\bar{y}_{i,t+1}) + \frac{n-B}{n}\nabla\psi_i(y_{i,t}).
\end{aligned}$$

Next, we bound the $\sharp$ term. When $\mathbf{y} = \mathbf{y}_*$, expectation of $\sharp$ is also zero which proves (5.36).

When $\mathbf{y}$ is possibly random, let us apply Lemma 3.13 to the update $\hat{y}_{i,t+1} = \arg\min_v -\Delta_{i,t}^\top v + \lambda_0 D_{\psi_i}(v, \hat{y}_{i,t}), \forall i$ (λ_0 to be determined), where

$$\Delta_{i,t} \coloneqq -\frac{n}{B}\nabla\psi_i(y_{i,t+1}) + \nabla\psi_i(\bar{y}_{i,t+1}) + \frac{n-B}{B}\nabla\psi_i(y_{i,t})$$

is a martingale sequence due to

$$\mathbb{E}_{\mathcal{B}_t}\left[(\nabla\psi_i(y_{i,t+1}) - \nabla\psi_i(y_{i,t}))\right] = \frac{B}{n}(\nabla\psi_i(\bar{y}_{i,t+1}) - \nabla\psi_i(y_{i,t})).$$

We have

$$\mathbb{E}[\sharp] \le \mathbb{E}\left[\frac{\lambda_0}{n}(D_{\psi_i}(y_i, \hat{y}_{i,t}) - D_{\psi_i}(y_i, \hat{y}_{i,t+1}))\right] + \frac{1}{2n\mu_\psi\lambda_0}\mathbb{E}\left[\left\|\Delta_{i,t}\right\|_2^2\right].$$

Note that $\mathbb{E}_{\mathcal{B}_t}\left[(\nabla\psi_i(y_{i,t+1}) - \nabla\psi_i(y_{i,t}))\right] = \frac{B}{n}(\nabla\psi_i(\bar{y}_{i,t+1}) - \nabla\psi_i(y_{i,t}))$ such that

$$\begin{aligned}
\mathbb{E}_{\mathcal{B}_t}\left[\left\|\Delta_{i,t}\right\|_2^2\right] &= \mathbb{E}_{\mathcal{B}_t}\left\|(\nabla\psi_i(\bar{y}_{i,t+1}) - \nabla\psi_i(y_{i,t})) - \frac{n}{B}(\nabla\psi_i(y_{i,t+1}) - \nabla\psi_i(y_{i,t})\right\|_2^2 \\
&\le \frac{n^2}{B^2}\mathbb{E}_{\mathcal{B}_t}\left\|\nabla\psi_i(y_{i,t+1}) - \nabla\psi_i(y_{i,t})\right\|_2^2 - \left\|(\nabla\psi_i(\bar{y}_{i,t+1}) - \nabla\psi_i(y_{i,t}))\right\|_2^2 \\
&\le \frac{n}{B}\left\|\nabla\psi_i(\bar{y}_{i,t+1}) - \nabla\psi_i(y_{i,t})\right\|_2^2 - \left\|(\nabla\psi_i(\bar{y}_{i,t+1}) - \nabla\psi_i(y_{i,t}))\right\|_2^2.
\end{aligned}$$

Thus, we have

$$\begin{aligned}
\mathbb{E}[\sharp] \le{}& \mathbb{E}\left[\frac{\lambda_0}{n}(D_{\psi_i}(y_i, \hat{y}_{i,t}) - D_{\psi_i}(y_i, \hat{y}_{i,t+1}))\right] \\
&+ \frac{n-B}{2\mu_\psi\lambda_0 nB}\mathbb{E}\left[\left\|\nabla\psi_i(\bar{y}_{i,t+1}) - \nabla\psi_i(y_{i,t})\right\|_2^2\right].
\end{aligned}$$

Averaging (5.38) multiplied by $\tau_t + \rho$ and combining (5.37) finishes the proof. □

Lemma 5.17 *Suppose ψ_i is μ_ψ-strongly convex. For any $\lambda_2, \lambda_3, \lambda_4, \lambda_5 > 0$ and $\mathbf{y}$ that possibly depends on all randomness in the algorithm, we have*

$$\begin{aligned}
\mathbb{E}[B_t(\mathbf{y})] = \frac{1}{n}\sum_{i=1}^n \mathbb{E}(g_i(\mathbf{w}_{t+1}) - \tilde{g}_{i,t})^\top(y_i - \bar{y}_{i,t+1}) &\le \mathbb{E}[\Gamma_{t+1} - \theta\Gamma_t] \qquad (5.39)\\
+ \frac{(\lambda_3 + \lambda_4\theta)\mathbb{E}[D_\psi(\bar{\mathbf{y}}_{t+1}, \mathbf{y}_t)]}{\mu_\psi n} &+ \frac{G_2^2\mathbb{E}\left\|\mathbf{w}_{t+1} - \mathbf{w}_t\right\|_2^2}{2\lambda_3} + \frac{G_2^2\theta\mathbb{E}\left\|\mathbf{w}_t - \mathbf{w}_{t-1}\right\|_2^2}{2\lambda_4} \\
+ \frac{(1 + 3.5\theta + 3.5\theta^2)\sigma_0^2\alpha_t}{\mu_\psi} &+ \frac{(1+\theta)\lambda_2\sigma_0^2}{2\mu_\psi} + \frac{\theta\sigma_0^2\lambda_5}{2\mu_\psi} \\
+ \frac{1+\theta}{n\lambda_2}\mathbb{E}[D_\psi(\mathbf{y}, \tilde{\mathbf{y}}_t) - D_\psi(\mathbf{y}, \tilde{\mathbf{y}}_{t+1})] &+ \frac{\theta}{n\lambda_5}\mathbb{E}[D_\psi(\mathbf{y}, \check{\mathbf{y}}_t) - D_\psi(\mathbf{y}, \check{\mathbf{y}}_{t+1})],
\end{aligned}$$

where $\Gamma_t \coloneqq \frac{1}{n}\sum_{i=1}^n (g_i(\mathbf{w}_t) - g_i(\mathbf{w}_{t-1}))^\top (y_i - y_{i,t})$ *and* $\check{\mathbf{y}}_t, \tilde{\mathbf{y}}_t$ *are some virtual sequences. In addition, we have*

$$\mathbb{E}[B_t(\mathbf{y}_*)] = \frac{1}{n}\sum_{i=1}^n \mathbb{E}(g_i(\mathbf{w}_{t+1}) - \tilde{g}_{i,t})^\top (y_{i,*} - \bar{y}_{i,t+1}) \le \mathbb{E}[\Gamma^*_{t+1} - \theta\Gamma^*_t] \tag{5.40}$$
$$+ \frac{(\lambda_3 + \lambda_4\theta)\mathbb{E}[D_\psi(\bar{\mathbf{y}}_{t+1}, \mathbf{y}_t)]}{\mu_\psi n} + \frac{G_2^2 \mathbb{E}\,\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2}{2\lambda_3} + \frac{G_2^2\theta\mathbb{E}\,\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2}{2\lambda_4}$$
$$+ \frac{(1 + 3.5\theta + 3.5\theta^2)\sigma_0^2\alpha_t}{\mu_\psi},$$

where $\Gamma^*_t \coloneqq \frac{1}{n}\sum_{i=1}^n (g_i(\mathbf{w}_t) - g_i(\mathbf{w}_{t-1}))^\top (y_{i,*} - y_{i,t})$.

Proof. Since

$$\tilde{g}_{i,t} = g_i(\mathbf{w}_t; \zeta_{i,t}) + \theta(g_i(\mathbf{w}_t; \zeta_{i,t}) - g_i(\mathbf{w}_{t-1}; \zeta_{i,t})),$$

we have

$$\begin{aligned}
&\frac{1}{n}\sum_{i=1}^n (g_i(\mathbf{w}_{t+1}) - \tilde{g}_{i,t})^\top (y_i - \bar{y}_{i,t+1}) \\
&= \underbrace{\frac{1+\theta}{n}\sum_{i=1}^n (g_i(\mathbf{w}_t) - g_i(\mathbf{w}_t; \zeta_{i,t}))^\top (y_i - \bar{y}_{i,t+1})}_{\text{I}} \\
&+ \underbrace{\frac{\theta}{n}\sum_{i=1}^n (g_i(\mathbf{w}_{t-1}; \zeta_{i,t}) - g_i(\mathbf{w}_{t-1}))^\top (y_i - \bar{y}_{i,t+1})}_{\text{II}} \\
&+ \underbrace{\frac{1}{n}\sum_{i=1}^n (g_i(\mathbf{w}_{t+1}) - g_i(\mathbf{w}_t)) + \theta(g_i(\mathbf{w}_{t-1}) - g_i(\mathbf{w}_t))^\top (y_i - \bar{y}_{i,t+1})}_{\text{III}}.
\end{aligned} \tag{5.41}$$

Define

$$\dot{y}_{i,t+1} \coloneqq \arg\max_{v \in \mathcal{Y}_i} \{v^\top((1+\theta)g_i(\mathbf{w}_t) - \theta g_i(\mathbf{w}_{t-1})) - f_i^*(v) - \frac{1}{\alpha_t} D_{\psi_i}(v, y_{i,t})\}, \forall i \in [n].$$

This update differs from that of $\bar{y}_{i,t+1}$ in that it uses full gradients instead of stochastic gradients. We decompose the I term in (5.41) as

$$
\begin{aligned}
\mathrm{I} &= \frac{1+\theta}{n}\sum_{i=1}^{n}(g_i(\mathbf{w}_t) - g_i(\mathbf{w}_t;\zeta_{i,t}))^\top(y_i - \bar{y}_{i,t+1}) \\
&= \underbrace{\frac{1+\theta}{n}\sum_{i=1}^{n}(g_i(\mathbf{w}_t) - g_i(\mathbf{w}_t;\zeta_{i,t}))^\top(\dot{y}_{i,t+1} - \bar{y}_{i,t+1})}_{\mathrm{I}_1} \\
&+ \underbrace{\frac{1+\theta}{n}\sum_{i=1}^{n}(g_i(\mathbf{w}_t) - g_i(\mathbf{w}_t;\zeta_{i,t}))^\top y_i}_{\mathrm{I}_2} - \underbrace{\frac{1+\theta}{n}\sum_{i=1}^{n}(g_i(\mathbf{w}_t) - g_i(\mathbf{w}_t;\zeta_{i,t}))^\top \dot{y}_{i,t+1}}_{\mathrm{I}_3}.
\end{aligned}
$$

Taking expectation over $\zeta_{i,t}, \forall i$ will make $\mathbb{E}_{\zeta_t}[\mathrm{I}_3] = 0$. Below, we will bound I_1 and I_2.

$$
\mathrm{I}_1 \le \frac{1+\theta}{n}\sum_{i=1}^{n}\left\|g_i(\mathbf{w}_t) - g_i(\mathbf{w}_t;\zeta_{i,t})\right\|_2 \left\|\dot{y}_{i,t+1} - \bar{y}_{i,t+1}\right\|_2.
$$

Since $D_{\psi_i}(y_i, y_{i,t})$ is μ_ψ-strongly convex, Lemma 3.8 implies that

$$
\begin{aligned}
&\|\dot{y}_{i,t+1} - \bar{y}_{i,t+1}\|_2 \\
&\le \frac{\alpha_t}{\mu_\psi}\left((1+\theta)\left\|g_i(\mathbf{w}_t) - g_i(\mathbf{w}_t;\zeta_{i,t})\right\|_2 + \theta\left\|g_i(\mathbf{w}_{t-1}) - g_i(\mathbf{w}_{t-1};\zeta_{i,t})\right\|_2\right)
\end{aligned}
$$

Hence

$$
\begin{aligned}
&\mathbb{E}_{\zeta_t}[\mathrm{I}_1] \le \frac{(1+\theta)\alpha_t}{n\mu_\psi}\cdot \\
&\sum_{i=1}^{n}\mathbb{E}\left[(1+1.5\theta)\left\|g_i(\mathbf{w}_t) - g_i(\mathbf{w}_t;\zeta_{i,t})\right\|_2^2 + 0.5\theta\left\|g_i(\mathbf{w}_{t-1}) - g_i(\mathbf{w}_{t-1};\zeta_{i,t})\right\|_2^2\right] \\
&\le \frac{(1+\theta)(1+2\theta)\sigma_0^2\alpha_t}{\mu_\psi}. \qquad (5.42)
\end{aligned}
$$

Next, let us handle I_2. Let us define an auxiliary sequence $\{\tilde{\mathbf{y}}_t\}_{t\ge 1}$,

$$
\tilde{y}_{i,t+1} = \arg\min_{v\in\mathcal{Y}_i}\{(g_i(\mathbf{w}_t;\zeta_{i,t}) - g_i(\mathbf{w}_t))^\top v + \frac{1}{\lambda_2}D_{\psi_i}(v, \tilde{y}_{i,t})\},
$$

where $\lambda_2 > 0$. Lemma 3.13 implies that

$$
(g_i(\mathbf{w}_t) - g_i(\mathbf{w}_t;\zeta_{i,t}))^\top y_i \le \frac{1}{\lambda_2}\mathbb{E}[D_{\psi_i}(y_i, \tilde{y}_{i,t}) - D_{\psi_i}(y_i, \tilde{y}_{i,t+1})] + \frac{\lambda_2\sigma_0^2}{2\mu_\psi}.
$$

Averaging over $i = 1, \ldots, n$ and multiplying $(1+\theta)$ yields a bound of I_2:

$$\mathbb{E}[\mathrm{I}_2] \leq \frac{1+\theta}{n\lambda_2}\mathbb{E}[D_\psi(\mathbf{y}, \tilde{\mathbf{y}}_t) - D_\psi(\mathbf{y}, \tilde{\mathbf{y}}_{t+1})] + \frac{(1+\theta)\lambda_2\sigma_0^2}{2\mu_\psi}.$$

As a result, the I term in (5.41) can be bounded as

$$\mathbb{E}[\mathrm{I}] \leq \frac{1+\theta}{n\lambda_2}\mathbb{E}[D_\psi(\mathbf{y}, \tilde{\mathbf{y}}_t) - D_\psi(\mathbf{y}, \tilde{\mathbf{y}}_{t+1})] + \frac{(1+\theta)\lambda_2\sigma_0^2}{2\mu_\psi} + \frac{(1+\theta)(1+2\theta)\sigma_0^2\alpha_t}{\mu_\psi}. \tag{5.43}$$

Similarly, the II term in (5.41) can be bounded as

$$\mathbb{E}[\mathrm{II}] \leq \frac{\theta}{n\lambda_5}\mathbb{E}[D_\psi(\mathbf{y}, \check{\mathbf{y}}_t) - D_\psi(\mathbf{y}, \check{\mathbf{y}}_{t+1})] + \frac{\theta\lambda_5\sigma_0^2}{2\mu_\psi} + \frac{\theta(0.5+1.5\theta)\sigma_0^2\alpha_t}{\mu_\psi}. \tag{5.44}$$

where

$$\check{y}_{i,t+1} = \arg\min_{v\in\mathcal{Y}_i}\{(g_i(\mathbf{w}_{t-1}) - g_i(\mathbf{w}_{t-1}; \zeta_{i,t}))^\top v + \lambda_5 D_{\psi_i}(v, \check{y}_{i,t})\}, \forall i.$$

Next, let us bound III. Recall $\Gamma_t \coloneqq \frac{1}{n}\sum_{i=1}^n (g_i(\mathbf{w}_t) - g_i(\mathbf{w}_{t-1}))^\top (y_i - y_{i,t})$. For any $\lambda_3, \lambda_4 > 0$, III can be rewritten as

$$\begin{aligned}
\mathrm{III} &= \Gamma_{t+1} - \theta\Gamma_t + \frac{1}{n}\sum_{i=1}^n (g_i(\mathbf{w}_{t+1}) - g_i(\mathbf{w}_t))^\top (y_{i,t+1} - \bar{y}_{i,t+1}) \\
&- \frac{\theta}{n}\sum_{i=1}^n (g_i(\mathbf{w}_t) - g_i(\mathbf{w}_{t-1}))^\top (y_{i,t} - \bar{y}_{i,t+1}) \\
&\leq \Gamma_{t+1} - \theta\Gamma_t + \frac{G_2^2\left\|\mathbf{w}_{t+1} - \mathbf{w}_t\right\|_2^2}{2\lambda_3} + \frac{\lambda_3\left\|\mathbf{y}_{t+1} - \bar{\mathbf{y}}_{t+1}\right\|_2^2}{2n} \\
&+ \frac{G_2^2\theta\left\|\mathbf{w}_t - \mathbf{w}_{t-1}\right\|_2^2}{2\lambda_4} + \frac{\lambda_4\theta\left\|\mathbf{y}_t - \bar{\mathbf{y}}_{t+1}\right\|_2^2}{2n}.
\end{aligned}$$

Note that $y_{i,t+1} = \bar{y}_{i,t+1}$ if $i \in \mathcal{B}_t$ and $y_{i,t+1} = y_{i,t}$ otherwise. Then, $\left\|\mathbf{y}_{t+1} - \bar{\mathbf{y}}_{t+1}\right\|_2^2 \leq \left\|\mathbf{y}_t - \bar{\mathbf{y}}_{t+1}\right\|_2^2$ such that

$$\begin{aligned}
\mathrm{III} \leq \Gamma_{t+1} - \theta\Gamma_t + \frac{G_2^2\left\|\mathbf{w}_{t+1} - \mathbf{w}_t\right\|_2^2}{2\lambda_3} + \frac{G_2^2\theta\left\|\mathbf{w}_t - \mathbf{w}_{t-1}\right\|_2^2}{2\lambda_4} \\
+ \frac{(\lambda_3 + \lambda_4\theta)D_\psi(\bar{\mathbf{y}}_{t+1}, \mathbf{y}_t)}{\mu_\psi n}.
\end{aligned} \tag{5.45}$$

Combining (5.43), (5.45), (5.44), we have

$$
\begin{aligned}
&\mathbb{E}[B_t(\mathbf{y})] \le \mathbb{E}[\Gamma_{t+1} - \theta\Gamma_t] \\
&+ \frac{1+\theta}{n\lambda_2}\mathbb{E}[D_\psi(\mathbf{y}, \tilde{\mathbf{y}}_t) - D_\psi(\mathbf{y}, \tilde{\mathbf{y}}_{t+1})] + \frac{\theta}{n\lambda_5}\mathbb{E}[D_\psi(\mathbf{y}, \check{\mathbf{y}}_t) - D_\psi(\mathbf{y}, \check{\mathbf{y}}_{t+1})] \\
&+ \frac{(\lambda_3 + \lambda_4\theta)\mathbb{E}[D_\psi(\bar{\mathbf{y}}_{t+1}, \mathbf{y}_t)]}{\mu_\psi n} + \frac{G_2^2\mathbb{E}\,\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2}{2\lambda_3} + \frac{G_2^2\theta\mathbb{E}\,\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2}{2\lambda_4} \\
&+ \frac{(1+\theta)\lambda_2\sigma_0^2}{2\mu_\psi} + \frac{\theta\sigma_0^2\lambda_5}{2\mu_\psi} + \frac{(1 + 3.5\theta + 3.5\theta^2)\sigma_0^2\alpha_t}{\mu_\psi}.
\end{aligned}
$$

□

Lemma 5.18 *When $\theta = 0$, for any $\lambda_2, \lambda_4 \ge 0$ and $\mathbf{y}$ that possibly depends on all randomness in the algorithm, we have*

$$
\mathbb{E}[B_t(\mathbf{y})] = \frac{1}{n}\sum_{i=1}^n \mathbb{E}(g_i(\mathbf{w}_{t+1}) - \tilde{g}_t)^\top(y_i - \bar{y}_{i,t+1}) \le \frac{G_2^2\mathbb{E}\left[\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2\right]}{4\lambda_4} + 4\lambda_4 G_1^2
$$

$$
+ \frac{1}{n\lambda_2}\mathbb{E}[D_\psi(\mathbf{y}, \tilde{\mathbf{y}}_t) - D_\psi(\mathbf{y}, \tilde{\mathbf{y}}_{t+1})] + \frac{\lambda_2\sigma_0^2}{2\mu_\psi} + \frac{\sigma_0^2\alpha_t}{\mu_\psi}. \tag{5.46}
$$

Proof. For ALEXR with $\theta = 0$, we have $\tilde{g}_{i,t} = g_i(\mathbf{w}_t; \zeta_{i,t})$. Then, for any $\lambda_4 > 0$ we have

$$
\frac{1}{n}\sum_{i=1}^n \mathbb{E}(g_i(\mathbf{w}_{t+1}) - \tilde{g}_t)^\top(y_i - \bar{y}_{i,t+1}) = \frac{1}{n}\sum_{i=1}^n \mathbb{E}\left[(g_i(\mathbf{w}_{t+1}) - g_i(\mathbf{w}_t))^\top(y_i - \bar{y}_{i,t+1})\right]
$$

$$
+ \frac{1}{n}\sum_{i=1}^n \mathbb{E}\left[(g_i(\mathbf{w}_t) - g_i(\mathbf{w}_t; \zeta_{i,t}))^\top(y_i - \bar{y}_{i,t+1})\right]. \tag{5.47}
$$

We bound the first term on the RHS by Young's inequality:

$$
\begin{aligned}
&\frac{1}{n}\sum_{i=1}^n \mathbb{E}\left[(g_i(\mathbf{w}_{t+1}) - g_i(\mathbf{w}_t))^\top(y_i - \bar{y}_{i,t+1})\right] \\
&\le \frac{1}{n}\sum_{i=1}^n \left(\frac{G_2^2\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2}{4\lambda_4} + \lambda_4\left\|y_i - \bar{y}_{i,t+1}\right\|_2^2\right) \le \frac{G_2^2\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2}{4\lambda_4} + 4G_1^2.
\end{aligned}
$$

The second term in (5.47) can be bounded similarly as (5.43) by:

$$
\begin{aligned}
&\frac{1}{n}\sum_{i=1}^n \mathbb{E}\left[(g_i(\mathbf{w}_t) - g_i(\mathbf{w}_t; \zeta_{i,t}))^\top(y_i - \bar{y}_{i,t+1})\right] \\
&\le \frac{1}{n\lambda_2}\mathbb{E}[D_\psi(\mathbf{y}, \tilde{\mathbf{y}}_t) - D_\psi(\mathbf{y}, \tilde{\mathbf{y}}_{t+1})] + \frac{\lambda_2\sigma_0^2}{2\mu_\psi} + \frac{\sigma_0^2\alpha_t}{\mu_\psi}.
\end{aligned}
$$

Combining the above inequalities together, we have

$$\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}(g_i(\mathbf{w}_{t+1})-\tilde{g}_t)^\top(y_i-\bar{y}_{i,t+1}) \le \frac{G_2^2\mathbb{E}\left[\|\mathbf{w}_{t+1}-\mathbf{w}_t\|_2^2\right]}{4\lambda_4}+4\lambda_4 G_1^2$$
$$+\frac{1}{n\lambda_2}\mathbb{E}[D_\psi(\mathbf{y},\tilde{\mathbf{y}}_t)-D_\psi(\mathbf{y},\tilde{\mathbf{y}}_{t+1})]+\frac{\lambda_2\sigma_0^2}{2\mu_\psi}+\frac{\sigma_0^2\alpha_t}{\mu_\psi}.$$

□

Lemma 5.19 *If g_i is L_2-smooth and $\eta \le \frac{1}{2G_1L_2}$, then*

$$\mathbb{E}[C_t(\mathbf{w}_*)] = \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}(g_i(\mathbf{w}_{t+1})-g_i(\mathbf{w}_*))^\top\bar{y}_{i,t+1}-\mathbf{z}_t^\top(\mathbf{w}_{t+1}-\mathbf{w}_*)\right] \tag{5.48}$$
$$\le \eta\sigma^2+\frac{1}{4\eta}\|\mathbf{w}_{t+1}-\mathbf{w}_t\|_2^2.$$

If g_i is G_2-Lipschitz continuous, then

$$\mathbb{E}[C_t(\mathbf{w}_*)] = \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}(g_i(\mathbf{w}_{t+1})-g_i(\mathbf{w}_*))^\top\bar{y}_{i,t+1}-\mathbf{z}_t^\top(\mathbf{w}_{t+1}-\mathbf{w}_*)\right] \tag{5.49}$$
$$\le \eta(\sigma^2+4G_1^2G_2^2)+\frac{1}{4\eta}\|\mathbf{w}_{t+1}-\mathbf{w}_t\|_2^2.$$

where $\sigma^2 = \frac{G_1^2\sigma_2^2}{B}+\frac{G_1^2G_2^2(n-B)}{B(n-1)}$.

Proof. We define $\Delta_t \coloneqq \frac{1}{B}\sum_{i\in\mathcal{B}_t}[\partial g_i(\mathbf{w}_t;\zeta'_{i,t})]^\top y_{i,t+1}-\frac{1}{n}\sum_{i=1}^{n}[\partial g_i(\mathbf{w}_t)]^\top\bar{y}_{i,t+1}$. Similar to Lemma 5.2, we have $\mathbb{E}_t[\|\Delta_t\|_2^2]\le\sigma^2$. To proceed, we have

$$\frac{1}{n}\sum_{i=1}^{n}(g_i(\mathbf{w}_{t+1})-g_i(\mathbf{w}_*))^\top\bar{y}_{i,t+1}-\mathbf{z}_t^\top(\mathbf{w}_{t+1}-\mathbf{w}_*)$$
$$=\frac{1}{n}\sum_{i=1}^{n}(g_i(\mathbf{w}_{t+1})-g_i(\mathbf{w}_t))^\top\bar{y}_{i,t+1}+\frac{1}{n}\sum_{i=1}^{n}(g_i(\mathbf{w}_t)-g_i(\mathbf{w}_*))^\top\bar{y}_{i,t+1}$$
$$+\frac{1}{n}\sum_{i=1}^{n}([\partial g_i(\mathbf{w}_t)]^\top\bar{y}_{i,t+1}+\Delta_t)^\top(\mathbf{w}_*-\mathbf{w}_{t+1}).$$

Since g_i is convex and $\mathcal{Y}_i\subset\mathbb{R}_+^n$ as $\partial f_i\ge 0$, we have

$$\frac{1}{n}\sum_{i=1}^{n}(g_i(\mathbf{w}_t)-g_i(\mathbf{w}_*))^\top\bar{y}_{i,t+1}\le\frac{1}{n}\sum_{i=1}^{n}[\nabla g_i(\mathbf{w}_t)]^\top(\mathbf{w}_t-\mathbf{w}_*)^\top\bar{y}_{i,t+1}.$$

Adding the above two inequalities, we have

$$\frac{1}{n}\sum_{i=1}^{n}(g_i(\mathbf{w}_{t+1}) - g_i(\mathbf{w}_*))^\top \bar{y}_{i,t+1} - \mathbf{z}_t^\top(\mathbf{w}_{t+1} - \mathbf{w}_*) \tag{5.50}$$
$$\leq \frac{1}{n}\sum_{i=1}^{n}(g_i(\mathbf{w}_{t+1}) - g_i(\mathbf{w}_t) - \nabla g_i(\mathbf{w}_t)^\top(\mathbf{w}_{t+1} - \mathbf{w}_t))^\top \bar{y}_{i,t+1} + \frac{1}{n}\sum_{i=1}^{n}\Delta_t^\top(\mathbf{w}_* - \mathbf{w}_{t+1}).$$

If g_i is L_2-smooth, the first term in (5.50) can be bounded by

$$\frac{1}{n}\sum_{i=1}^{n}(g_i(\mathbf{w}_{t+1}) - g_i(\mathbf{w}_t) - \nabla g_i(\mathbf{w}_t)^\top(\mathbf{w}_{t+1} - \mathbf{w}_t))^\top \bar{y}_{i,t+1}$$
$$\leq \frac{G_1}{n}\sum_{i=1}^{n}\left\|g_i(\mathbf{w}_{t+1}) - g_i(\mathbf{w}_t) - \nabla g_i(\mathbf{w}_t)^\top(\mathbf{w}_{t+1} - \mathbf{w}_t)\right\|_2 \leq \frac{G_1 L_2}{2}\left\|\mathbf{w}_{t+1} - \mathbf{w}_t\right\|_2^2. \tag{5.51}$$

To bound the second term in (5.50), we note that $\mathbb{E}_{\mathcal{B}_t,\tilde{\zeta}_t}[\Delta_t] = 0$. Let us define $\hat{\mathbf{w}}_{t+1} = \arg\min_{\mathbf{w}} \mathbf{w}^\top \frac{1}{n}\sum_{i=1}^{n}[\nabla g_i(\mathbf{w}_t)]^\top \bar{y}_{i,t+1} + \frac{1}{2\eta}\|\mathbf{w} - \mathbf{w}_t\|_2^2 + r(\mathbf{w})$. Then we have

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\Delta_t^\top(\mathbf{w}_* - \mathbf{w}_{t+1})\right] = \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\Delta_t^\top(\mathbf{w}_* - \hat{\mathbf{w}}_{t+1} + \hat{\mathbf{w}}_{t+1} - \mathbf{w}_{t+1})\right]$$
$$= \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\Delta_t^\top(\hat{\mathbf{w}}_{t+1} - \mathbf{w}_{t+1})\right],$$

where we use the fact that

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\Delta_t^\top(\mathbf{w}_* - \hat{\mathbf{w}}_{t+1})\right] = \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}_{\mathcal{B}_t,\zeta_t'}[\Delta_t]^\top(\mathbf{w}_* - \hat{\mathbf{w}}_{t+1})\right] = 0.$$

According to Lemma 1.7 we have

$$\mathbb{E}[\Delta_t^\top(\hat{\mathbf{w}}_{t+1} - \mathbf{w}_{t+1})] \leq \frac{\eta}{1+\mu\eta}\mathbb{E}\left\|\Delta_t\right\|_2^2 \leq \frac{\eta\sigma^2}{1+\mu\eta}. \tag{5.52}$$

Then, combining (5.50), (5.51) and (5.52) leads to

$$\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[(g_i(\mathbf{w}_{t+1}) - g_i(\mathbf{w}_*))^\top \bar{y}_{i,t+1}] - \mathbb{E}\mathbf{z}_t^\top(\mathbf{w}_{t+1} - \mathbf{w}_*)$$
$$\leq \frac{\eta\sigma^2}{1+\mu\eta} + \frac{L_2 G_1}{2}\left\|\mathbf{w}_{t+1} - \mathbf{w}_t\right\|_2^2,$$

which finishes the first part by noting the condition on η.

If g_i is G_2-Lipschitz continuous, we have

$$\frac{G_1}{n}\sum_{i=1}^{n}\|g_i(\mathbf{w}_{t+1})-g_i(\mathbf{w}_t)-\partial g_i(\mathbf{w}_t)(\mathbf{w}_{t+1}-\mathbf{w}_t)\|_2 \tag{5.53}$$
$$\leq 2G_1G_2\|\mathbf{w}_{t+1}-\mathbf{w}_t\|_2 \leq \eta 4G_1^2G_2^2+\frac{1}{4\eta}\|\mathbf{w}_{t+1}-\mathbf{w}_t\|_2^2 .$$

Combining (5.50), (5.52), and (5.53), we get

$$\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[g_i(\mathbf{w}_{t+1})-g_i(\mathbf{w}_*)^\top\bar{y}_{i,t+1}]-\mathbb{E}\mathbf{z}_t^\top(\mathbf{w}_{t+1}-\mathbf{w}_*)$$
$$\leq \eta(\sigma^2+4G_1^2G_2^2)+\frac{1}{4\eta}\|\mathbf{w}_{t+1}-\mathbf{w}_t\|_2^2 .$$

□

5.4.3 Strongly convex objectives

In this section, we derive a complexity of $O(1/\epsilon)$ under the the following condition.

Assumption 5.11. *We assume that the function r is μ-strongly convex ($\mu > 0$) and each f_i is L_1-smooth, both with respect to the Euclidean norm $\|\cdot\|_2$.*

With this assumption, the minimax problem becomes strongly convex and strongly concave since the dual f_i^* is $1/L_1$-strongly convex with respect to $\|\cdot\|_2$. In this case, we will establish the convergence of $\mu\|\mathbf{w}-\mathbf{w}_*\|_2^2$.

Critical: Under Assumption 5.11, parts (i) and (ii) of Assumption 5.9 hold for both variants of ALEXR. For ALEXR-v1, we have $\mu_\psi = 1/L_1$ and $\rho = 1$, whereas for ALEXR-v2, we have $\mu_\psi = 1$ and $\rho = 1/L_1$. Hence, the following theorem holds for both variants of ALEXR.

Let us introduce a few notations:

$$a=\frac{\epsilon\mu_\psi\rho}{24\sigma_0^2},\quad b_1=3(\sigma^2+4G_1^2G_2^2),\quad b_2=3\sigma^2.$$

Theorem 5.7 *Suppose Assumptions 5.8, 5.10 and 5.11 hold.*

- *If g_i is G_2-Lipschitz continuous, by setting $\alpha=\frac{1-\theta}{\rho(\theta-(1-B/n))}$, $\eta=\frac{1-\theta}{\theta\mu}$ and*

$$\theta=\max\left\{1-\frac{a\frac{B}{n}}{1+a},\ 1-\frac{\mu\epsilon}{b_1+\mu\epsilon}\right\}.$$

ALEXR finds a solution $\mathbf{w}_{T+1}$ such that $\mathbb{E}[\mu\|\mathbf{w}_{T+1}-\mathbf{w}_*\|_2^2]\leq\epsilon$ with an iteration complexity of

$$T = O\left(\frac{1}{1-\theta}\log(3\Upsilon/\epsilon)\right) = \tilde{O}\left(\max\left(\frac{n}{B}, \frac{(\sigma^2 + G_1^2 G_2^2)}{\mu\epsilon}, \frac{n\sigma_0^2}{B\epsilon\mu_\psi\rho}\right)\right).$$

- *If g_i is further L_2-smooth, by setting $\alpha = \frac{1-\theta}{\rho(\theta-(1-B/n))}$, $\eta = \frac{1-\theta}{\theta\mu}$ and*

$$\theta = \max\left\{1 - \frac{a\frac{B}{n}}{1+a}, 1 - \frac{\mu\epsilon}{b_2+\mu\epsilon}, 1 - \frac{\mu}{2G_1L_2+\mu}\right\},$$

for sufficiently small ϵ, ALEXR finds a solution $\mathbf{w}_{T+1}$ such that $\mathbb{E}[\mu\|\mathbf{w}_{T+1} - \mathbf{w}_*\|_2^2] \leq \epsilon$ with an iteration complexity of

$$T = O\left(\frac{1}{1-\theta}\log(3\Upsilon/\epsilon)\right) = \tilde{O}\left(\max\left(\frac{G_1L_2}{\mu}, \frac{n}{B}, \frac{\sigma^2}{\mu\epsilon}, \frac{n\sigma_0^2}{B\epsilon\mu_\psi\rho}\right)\right).$$

where $\Upsilon = \frac{\mu}{2}\|\mathbf{w}_1 - \mathbf{w}_*\|_2^2 + \frac{2\rho}{B}D_\psi(\mathbf{y}_*, \mathbf{y}_1)$ and $\sigma^2 = \frac{G_1^2\sigma_2^2}{B} + \frac{G_1^2G_2^2(n-B)}{B(n-1)}$.

Why it matters

For smooth functions g_i, the iteration complexity is improved in the sense that the $O(1/\epsilon)$ dependence is scaled by the variance of the stochastic estimators. In contrast, for non-smooth g_i, the complexity always has a term $\frac{G_1^2G_2^2}{\mu\epsilon}$ independent of variance.

Proof. We first consider non-smooth g_i. Combining (5.32), (5.36) for $A_t(\mathbf{y}_*)$, (5.40) for $B_t(\mathbf{y}_*)$, (5.49) for $C_t(\mathbf{w}_*)$ together we have

$$\begin{aligned}
&\mathbb{E}[F(\mathbf{w}_{t+1}, \mathbf{y}_*) - F(\mathbf{w}_*, \bar{\mathbf{y}}_{t+1})] \\
&\leq \frac{1}{2\eta_t}\|\mathbf{w}_* - \mathbf{w}_t\|_2^2 - (\frac{1}{2\eta_t} + \frac{\mu}{2})\|\mathbf{w}_* - \mathbf{w}_{t+1}\|_2^2 - \frac{1}{2\eta_t}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 \\
&+ \mathbb{E}\left[\frac{\tau_t + \rho\left(1 - \frac{B}{n}\right)}{B}D_\psi(\mathbf{y}_*, \mathbf{y}_t) - \frac{\tau_t + \rho}{B}D_\psi(\mathbf{y}_*, \mathbf{y}_{t+1})\right] - \frac{\tau_t}{n}\mathbb{E}\left[D_\psi(\bar{\mathbf{y}}_{t+1}, \mathbf{y}_t)\right] \\
&+ \mathbb{E}[\Gamma_{t+1}^* - \theta\Gamma_t^*] + \frac{(\lambda_3 + \lambda_4\theta)\mathbb{E}[D_\psi(\bar{\mathbf{y}}_{t+1}, \mathbf{y}_t)]}{\mu_\psi n} \\
&+ \frac{G_2^2\mathbb{E}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2}{2\lambda_3} + \frac{G_2^2\theta\mathbb{E}\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2}{2\lambda_4} + \frac{(1 + 3.5\theta + 3.5\theta^2)\sigma_0^2\alpha_t}{\mu_\psi} \\
&+ \eta_t(\sigma^2 + 4G_1^2G_2^2) + \frac{1}{4\eta_t}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2.
\end{aligned}$$

Define $\Upsilon_{1,t} \coloneqq \frac{1}{2}\|\mathbf{w}_* - \mathbf{w}_t\|_2^2$ and $\Upsilon_{2,t} = \frac{1}{B}D_\psi(\mathbf{y}_*, \mathbf{y}_t)$. Since

$$F(\mathbf{w}_{t+1}, \mathbf{y}_*) - F(\mathbf{w}_*, \bar{\mathbf{y}}_{t+1}) \geq F(\mathbf{w}_{t+1}, \mathbf{y}_*) - F(\mathbf{w}_*, \mathbf{y}_*) + F(\mathbf{w}_*, \mathbf{y}_*) - F(\mathbf{w}_*, \bar{\mathbf{y}}_{t+1}) \geq 0,$$

multiplying the above inequality by θ^{-t} on both sides, we have

$$
\begin{aligned}
0 \leq\ & \theta^{-t}\mathbb{E}\left[\frac{1}{\eta_t}\Upsilon_{1,t} + (\tau_t + \rho(1-\frac{B}{n})))\Upsilon_{2,t} - \theta\Gamma_t^*\right] \\
& - \theta^{-t}\mathbb{E}\left[(\frac{1}{\eta_t}+\mu)\Upsilon_{1,t+1} + (\tau_t+\rho)\Upsilon_{2,t+1} - \Gamma_{t+1}^*\right] \\
& - \theta^{-t}(\frac{1}{2\eta_t}\mathbb{E}\left\|\mathbf{w}_{t+1}-\mathbf{w}_t\right\|_2^2 + \frac{\tau_t}{n}\mathbb{E}\left[D_\psi(\bar{\mathbf{y}}_{t+1},\mathbf{y}_t)\right]) + \theta^{-t}\frac{(\lambda_3+\lambda_4\theta)\mathbb{E}[D_\psi(\bar{\mathbf{y}}_{t+1},\mathbf{y}_t)]}{\mu_\psi n} \\
& + \theta^{-t}\frac{G_2^2\mathbb{E}\left\|\mathbf{w}_{t+1}-\mathbf{w}_t\right\|_2^2}{2\lambda_3} + \theta^{-t}\frac{G_2^2\theta\mathbb{E}\left\|\mathbf{w}_t-\mathbf{w}_{t-1}\right\|_2^2}{2\lambda_4} + \theta^{-t}\frac{1}{4\eta_t}\mathbb{E}\left\|\mathbf{w}_{t+1}-\mathbf{w}_t\right\|_2^2 \\
& + \theta^{-t}\left(\frac{(1+3.5\theta+3.5\theta^2)\sigma_0^2\alpha_t}{\mu_\psi} + \eta_t(\sigma^2+4G_1^2G_2^2)\right).
\end{aligned} \tag{5.54}
$$

Let

$$
\frac{1}{\eta_{t-1}} + \mu = \frac{1}{\eta_t\theta}, \quad (\tau_{t-1}+\rho) = \frac{1}{\theta}(\tau_t + \rho(1-\frac{B}{n})). \tag{5.55}
$$

Hence,

$$
\begin{aligned}
& \sum_{t=1}^{T}\left\{\theta^{-t}\left[\frac{1}{\eta_t}\Upsilon_{1,t} + (\tau_t+\rho(1-\frac{B}{n})))\Upsilon_{2,t} - \theta\Gamma_t^*\right]\right. \\
& \left. -\theta^{-t}\left[(\frac{1}{\eta_t}+\mu)\Upsilon_{1,t+1} + (\tau_t+\rho)\Upsilon_{2,t+1} - \Gamma_{t+1}^*\right]\right\} \\
& \leq \sum_{t=1}^{T}\left\{\theta^{-(t-1)}\left[(\frac{1}{\eta_{t-1}}+\mu)\Upsilon_{1,t} + (\tau_{t-1}+\rho)\Upsilon_{2,t} - \Gamma_t^*\right]\right. \\
& \left. -\theta^{-t}\left[(\frac{1}{\eta_t}+\mu)\Upsilon_{1,t+1} + (\tau_t+\rho)\Upsilon_{2,t+1} - \Gamma_{t+1}^*\right]\right\} \\
& = \left[(\frac{1}{\eta_0}+\mu)\Upsilon_{1,1} + (\tau_0+\rho)\Upsilon_{2,1} - \Gamma_1\right] \\
& - \theta^{-T}\left[(\frac{1}{\eta_T}+\mu)\Upsilon_{1,T+1} + (\tau_T+\rho)\Upsilon_{2,T+1} - \Gamma_{T+1}\right].
\end{aligned}
$$

Since

$$
\begin{aligned}
& -\Gamma_{T+1} \geq -\frac{1}{n}\sum_{i=1}^{n} G_2\|\mathbf{w}_{T+1}-\mathbf{w}_T\|_2\|y_{i,*}-y_{i,T+1}\|_2 \\
& \geq -\frac{1}{n}\sum_{i=1}^{n}(\frac{G_2^2 B}{n(\rho+\tau_T)\mu_\psi}\|\mathbf{w}_{T+1}-\mathbf{w}_T\|_2^2 + \frac{n\mu_\psi(\rho+\tau_T)}{4B}\|y_{i,*}-y_{i,T+1}\|_2^2) \\
& \geq -(\frac{G_2^2 B}{2n(\rho+\tau_T)\mu_\psi}\|\mathbf{w}_{T+1}-\mathbf{w}_T\|_2^2 + \frac{\rho+\tau_T}{2}\Upsilon_{2,T+1}).
\end{aligned} \tag{5.56}
$$

Summing (5.54) over $t = 1, \ldots, T$ and utilizing the above two inequalities, we have

$$\begin{aligned}
&\theta^{-T}\mathbb{E}\left[(\frac{1}{\eta_T}+\mu)\Upsilon_{1,T+1}+\frac{\rho+\tau_T}{2}\Upsilon_{2,T+1}\right]\\
&\leq\left[(\frac{1}{\eta_0}+\mu)\Upsilon_{1,1}+(\tau_0+\rho)\Upsilon_{2,1}-\Gamma_1\right]+\\
&\frac{\theta^{-T}G_2^2B}{2n(\rho+\tau_T)\mu_\psi}\mathbb{E}\|\mathbf{w}_{T+1}-\mathbf{w}_T\|_2^2-\mathbb{E}\left[\sum_{t=1}^{T}\frac{\theta^{-t}}{2}(\frac{1}{\eta_t}-\frac{G_2^2}{\lambda_3}-\frac{G_2^2}{\lambda_4}-\frac{1}{2\eta_t})\|\mathbf{w}_{t+1}-\mathbf{w}_t\|_2^2\right]\\
&-\mathbb{E}\left[\sum_{t=1}^{T}\frac{\theta^{-t}}{n}(\frac{1}{\alpha_t}-\frac{\lambda_3+\lambda_4\theta}{\mu_\psi})D_\psi(\bar{\mathbf{y}}_{t+1},\mathbf{y}_t)\right]\\
&+\sum_{t=1}^{T}\theta^{-t}\left(\frac{(1+3.5\theta+3.5\theta^2)\sigma_0^2\alpha_t}{\mu_\psi}+\eta_t(\sigma^2+4G_1^2G_2^2)\right),
\end{aligned}$$

where we use the fact $\sum_{t=1}^{T}\|\mathbf{w}_t-\mathbf{w}_{t-1}\|_2^2\leq\sum_{t=1}^{T+1}\|\mathbf{w}_t-\mathbf{w}_{t-1}\|_2^2=\sum_{t=1}^{T}\|\mathbf{w}_{t+1}-\mathbf{w}_t\|_2^2$.

Let $\eta_t=\eta,\alpha_t=\frac{1}{\tau_t}=\alpha,\lambda_3=\lambda_4=8\eta G_2^2$. If $\alpha\leq\frac{\mu_\psi}{16\eta G_2^2}$ (to be verfied later), we have $\tau\geq\frac{4\eta G_2^2B}{n\mu_\psi}$. As a result, $\frac{G_2^2B}{2n(\rho+\tau_t)\mu_\psi}\leq\frac{1}{8\eta}$ and $\frac{1}{\alpha_t}\geq\frac{16\eta G_2^2}{\mu_\psi}$. Then the terms related to $\|\mathbf{w}_{t+1}-\mathbf{w}_t\|$ and $D_\psi(\bar{\mathbf{y}}_{t+1},\mathbf{y}_t)$ is less than zero. As a result,

$$\begin{aligned}
&\left[(\frac{1}{\eta}+\mu)\Upsilon_{1,T+1}+(\frac{\rho}{2}+\frac{1}{2\alpha})\Upsilon_{2,T+1}\right]\\
&\leq\theta^T\left[(\frac{1}{\eta}+\mu)\Upsilon_{1,1}+(\frac{1}{\alpha}+\rho)\Upsilon_{2,1}\right]+\sum_{t=1}^{T}\theta^{T-t}\left(\frac{8\sigma_0^2\alpha}{\mu_\psi}+\eta(\sigma^2+4G_1^2G_2^2)\right)\\
&\leq\theta^T\left[(\frac{1}{\eta}+\mu)\Upsilon_{1,1}+(\frac{1}{\alpha}+\rho)\Upsilon_{2,1}\right]+\frac{1}{1-\theta}\left(\frac{8\sigma_0^2\alpha}{\mu_\psi}+\eta(\sigma^2+4G_1^2G_2^2)\right).
\end{aligned}$$

Due to the relationship between η,α and θ in (5.55), we have

$$\begin{aligned}
\theta&=\frac{1}{1+\mu\eta}=\frac{1+\alpha\rho(1-B/n)}{1+\alpha\rho}\geq\frac{1}{1+\alpha\rho}\\
\alpha&=\frac{1-\theta}{\rho(\theta-(1-B/n))},\quad\eta=\frac{1-\theta}{\theta\mu}.
\end{aligned}$$

Then, we have

$$
\begin{aligned}
\left[\mu\Upsilon_{1,T+1}\right] &= \frac{\mu\eta}{1+\eta\mu}\left(\frac{1}{\eta}+\mu\right)\Upsilon_{1,T+1} \\
&\le \theta^T\mu\left[\Upsilon_{1,1}+\frac{(1+\alpha\rho)\eta}{\alpha(1+\eta\mu)}\Upsilon_{2,1}\right]+\frac{1}{1-\theta}\frac{\eta\mu}{1+\eta\mu}\left(\frac{8\sigma_0^2\alpha}{\mu_\psi}+\eta(\sigma^2+4G_1^2G_2^2)\right) \\
&= \theta^T\Upsilon+\frac{8\sigma_0^2\alpha}{\mu_\psi}+\eta(\sigma^2+4G_1^2G_2^2) \\
&\le \theta^T\Upsilon+\frac{1-\theta}{\rho(\theta-(1-B/n))}\frac{8\sigma_0^2}{\mu_\psi}+\frac{1-\theta}{\theta\mu}(\sigma^2+4G_1^2G_2^2),
\end{aligned}
$$

where $\Upsilon = \mu\Upsilon_{1,1}+\mu\frac{(1+\alpha\rho)\eta}{\alpha(1+\eta\mu)}\Upsilon_{2,1}$.

To let the RHS be less than ϵ, it is sufficient to have

$$
\begin{aligned}
T &\ge \frac{1}{1-\theta}\log(3\Upsilon/\epsilon) \ge \frac{-1}{\log(\theta)}\log(3\Upsilon/\epsilon) \Rightarrow \theta^T\Upsilon \le \epsilon/3, \\
\theta &\ge 1-\frac{\epsilon\mu_\psi\rho B/(24\sigma_0^2 n)}{1+\epsilon\mu_\psi\rho/(24\sigma_0^2)} \Rightarrow \frac{1-\theta}{\rho(\theta-(1-B/n))}\frac{8\sigma_0^2}{\mu_\psi} \le \epsilon/3, \\
\theta &\ge \frac{1}{1+\mu\epsilon/(3(\sigma^2+4G_1^2G_2^2))} \Rightarrow \frac{1-\theta}{\theta\mu}(\sigma^2+4G_1^2G_2^2) \le \frac{\epsilon}{3}.
\end{aligned}
$$

As a result,

$$
T = O\left(\frac{1}{1-\theta}\log(3\Upsilon/\epsilon)\right) = \tilde{O}\left(\max\left(\frac{(\sigma^2+G_1^2G_2^2)}{\mu\epsilon}, \frac{n}{B}, \frac{n\sigma_0^2}{B\epsilon\mu_\psi\rho}\right)\right).
$$

Finally, we verify that if $\epsilon^2 \le \frac{9(\sigma^2+4G_1^2G_2^2)\sigma_0^2}{2G_2^2}$, then it holds that

$$
\alpha \le \frac{\mu_\psi\epsilon}{24\sigma_0^2} = \frac{\mu_\psi\epsilon^2}{24\sigma_0^2\epsilon} \le \frac{\mu_\psi 3(\sigma^2+4G_1^2G_2^2)}{16G_2^2\epsilon} \le \frac{\mu_\psi\theta\mu}{16G_2^2(1-\theta)} = \frac{\mu_\psi}{16\eta G_2^2}.
$$

Since $\alpha\rho \le O(1)$, we have

$$
\frac{(1+\alpha\rho)\eta}{(1+\mu\eta)\alpha} \le 2\frac{\eta}{\alpha} \le 2\frac{\rho}{\mu},
$$

thus $\Upsilon_{1,1}+\frac{(1+\alpha\rho)\eta}{\alpha(1+\mu\eta)}\Upsilon_{2,1} \le \Upsilon_{1,1}+\frac{2\rho}{\mu}\Upsilon_{2,1}$. Thus, $\Upsilon \le \mu\Upsilon_{1,1}+2\rho\Upsilon_{2,1}$.

For smooth g_i, the proof is similar by using (5.48) instead of using (5.49). Hence, $\eta_t(\sigma^2+4G_1^2G_2^2)$ becomes $\eta_t(\sigma^2)$ and there is additional condition $\eta_t \le \frac{1}{2G_1L_2}$, which transfers to a condition on θ. □

5.4.4 Convex objectives with non-smooth outer functions

In this section, we only consider ALEXR-v2 for solving convex objectives with non-smooth f_i. For ALEXR-v2, we have that ψ is 1-smooth and 1-strongly convex. Hence, we have

$$\frac{(n-B)(\tau+\rho)}{2\mu_\psi\lambda_0 nB}\mathbb{E}\left[\sum_{i=1}^n \left\|\nabla\psi_i(\bar{y}_{i,t+1}) - \nabla\psi_i(y_{i,t})\right\|_2^2\right] \tag{5.57}$$

$$\leq \frac{(n-B)(\tau+\rho)}{2\lambda_0 nB}\mathbb{E}\left[\sum_{i=1}^n \left\|\bar{y}_{i,t+1} - y_{i,t}\right\|_2^2\right] \leq \frac{(n-B)(\tau+\rho)}{\lambda_0 nB}\mathbb{E}\left[D_\psi(\bar{\mathbf{y}}_{t+1}, \mathbf{y}_t)\right].$$

Theorem 5.8 *Suppose Assumption 5.9 holds with $\rho = 0, \mu_\psi = 1$, and Assumptions 5.8, 5.10 hold. If g_i is G_2-Lipschitz continuous, setting $\theta = 0$ and*

$$\alpha = \frac{\epsilon}{6\sigma_0^2}, \quad \eta = \frac{\epsilon}{6(\sigma^2 + 8G_1^2G_2^2)},$$

ALEXR-v2 returns an ϵ-optimal solution $\bar{\mathbf{w}}_T = \sum_{t=1}^T \mathbf{w}_t/T$ in expectation with a complexity of

$$T = O\left(\frac{\sigma^2 + G_1^2G_2^2}{\epsilon^2}, \frac{\Omega\sigma_0^2}{B\epsilon^2}, \frac{\Omega\sigma_0^2}{n\epsilon^2}\right).$$

where Ω is a constant such that $\mathbb{E}[D_\psi(\mathbf{y}_T^*, \mathbf{y}_1)] \leq \Omega \leq O(G_1^2 n)$, and $\mathbf{y}_T^* = \arg\max_{\mathbf{y}\in\mathcal{Y}_1\times\cdots\times\mathcal{Y}_n} F(\bar{\mathbf{w}}_T, \mathbf{y})$.

Why it matters

In the worst case, the complexity is $O\left(\frac{G_1^2G_2^2}{\epsilon^2} + \frac{nG_1^2\sigma_0^2}{B\epsilon^2}\right)$. This will match the lower bounds established in next section.

Proof. Combining (5.35) with (5.57) yields

$$\mathbb{E}\left[\sum_{t=1}^T A_t(\mathbf{y})\right] \leq \frac{\tau}{B}\mathbb{E}[D_\psi(\mathbf{y}, \mathbf{y}_1)] - \frac{\tau}{n}\mathbb{E}\left[\sum_{t=1}^T D_\psi(\bar{\mathbf{y}}_{t+1}, \mathbf{y}_t)\right] + \frac{\lambda_0\tau}{n}\mathbb{E}D_\psi(\mathbf{y}, \hat{\mathbf{y}}_1) \tag{5.58}$$

$$+ \frac{(n-B)\tau}{\lambda_0 nB}\mathbb{E}\left[\sum_{t=1}^T D_\psi(\bar{\mathbf{y}}_{t+1}, \mathbf{y}_t)\right] \tag{5.59}$$

Adding this inequality with (5.32), (5.46), and (5.49) over $t = 1, \ldots, T$, we have

$$\mathbb{E}\left[\sum_{t=1}^{T} F(\mathbf{w}_{t+1}, \mathbf{y}) - F(\mathbf{w}_*, \bar{\mathbf{y}}_{t+1})\right] \leq \frac{1}{2\eta}\|\mathbf{w}_* - \mathbf{w}_1\|_2^2 - \frac{1}{2\eta}\sum_{t=1}^{T}\mathbb{E}[\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2]$$
$$+ \frac{\tau}{B}\mathbb{E}[D_\psi(\mathbf{y}, \mathbf{y}_1)] - \frac{\tau}{n}\mathbb{E}\left[\sum_{t=1}^{T} D_\psi(\bar{\mathbf{y}}_{t+1}, \mathbf{y}_t)\right] + \frac{\lambda_0\tau}{n}\mathbb{E}D_\psi(\mathbf{y}, \hat{\mathbf{y}}_1)$$
$$+ \frac{(n-B)\tau}{\lambda_0 nB}\mathbb{E}\left[\sum_{t=1}^{T} D_\psi(\bar{\mathbf{y}}_{t+1}, \mathbf{y}_t)\right],$$
$$+ \frac{G_2^2}{4\lambda_4}\mathbb{E}\left[\sum_{t=1}^{T}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2\right] + 4\lambda_4 T G_1^2 + \frac{1}{n\lambda_2}\mathbb{E}[D_\psi(\mathbf{y}, \tilde{\mathbf{y}}_1)]$$
$$+ \frac{\lambda_2\sigma_0^2}{2}T + \sigma_0^2\alpha T,$$
$$+ \eta T(\sigma^2 + 4G_1^2G_2^2) + \frac{1}{4\eta}\sum_{t=1}^{T}\mathbb{E}\,\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2\,.$$

If we set $\lambda_0 = \frac{n-B}{B}$ and $\frac{G_2^2}{4\lambda_4} = \frac{1}{4\eta}$, we observe that the terms involving $\mathbb{E}\left[\sum_{t=1}^{T}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|^2\right]$ and $\mathbb{E}\left[\sum_{t=1}^{T} D_\psi(\bar{\mathbf{y}}_{t+1}, \mathbf{y}_t)\right]$ cancel out, leaving us with the following:

$$\mathbb{E}\left[\sum_{t=1}^{T} F(\mathbf{w}_{t+1}, \mathbf{y}) - F(\mathbf{w}_*, \bar{\mathbf{y}}_{t+1})\right]$$
$$\leq \frac{1}{2\eta}\|\mathbf{w}_* - \mathbf{w}_1\|_2^2 + \left(\frac{\tau(1+\lambda_0 B/n)}{B} + \frac{1}{n\lambda_2}\right)\mathbb{E}D_\psi(\mathbf{y}, \mathbf{y}_1)$$
$$+ \eta T(\sigma^2 + 8G_1^2G_2^2) + \frac{\lambda_2\sigma_0^2}{2}T + \sigma_0^2\alpha T.$$

Let $\mathbf{y} = \mathbf{y}_T^* = \arg\max F(\bar{\mathbf{w}}_T, \mathbf{y})$. Since $\frac{1}{T}\sum_{t=1}^{T} F(\mathbf{w}_{t+1}, \mathbf{y}) \geq F(\bar{\mathbf{w}}_T, \mathbf{y}_T^*) = F(\bar{\mathbf{w}}_T)$ and $F(\mathbf{w}_*, \bar{\mathbf{y}}_{t+1}) \leq F(\mathbf{w}_*, \mathbf{y}_*)$, we have

$$\mathbb{E}\left[F(\bar{\mathbf{w}}_T) - F(\mathbf{w}_*)\right] \leq \frac{1}{2\eta T}\|\mathbf{w}_* - \mathbf{w}_1\|_2^2 + \frac{1}{T}\left(\frac{\tau(1+\lambda_0 B/n)}{B} + \frac{1}{n\lambda_2}\right)\Omega$$
$$+ \eta(\sigma^2 + 8G_1^2G_2^2) + \frac{\lambda_2\sigma_0^2}{2} + \sigma_0^2\alpha. \tag{5.60}$$

Let

$$\alpha = \frac{\epsilon}{6\sigma_0^2}, \quad \lambda_2 = \frac{\epsilon}{3\sigma_0^2}, \quad \eta = \frac{\epsilon}{6(\sigma^2 + 8G_1^2G_2^2)},$$
$$T \geq O\left(\max\left(\frac{\|\mathbf{w}_1 - \mathbf{w}_*\|_2^2}{12\eta\epsilon}, \frac{\Omega(1+\lambda_0 B/n)}{6B\epsilon\alpha}, \frac{\Omega}{6n\lambda_2\epsilon}\right)\right).$$

Then, the RHS of (5.60) is less than ϵ. As a result, the complexity is in the order of

$$O\left(\max\left(\frac{\sigma^2+G_1^2G_2^2}{\epsilon^2},\frac{\Omega\sigma_0^2}{B\epsilon^2},\frac{\Omega\sigma_0^2}{n\epsilon^2}\right)\right).$$

□

Theorem 5.9 *Suppose Assumption 5.9 holds with $\rho=0,\mu_\psi=1$, Assumptions 5.8, 5.10 hold. If g_i is G_2-Lipschitz continuous and L_2-smooth, for sufficiently small ϵ, setting $\theta=1$ and*

$$\alpha=\frac{\epsilon}{64\sigma_0^2},\quad \eta=\min\left(\frac{\epsilon}{8\sigma^2},\frac{1}{2G_1L_2}\right)$$

ALEXR-v2 returns an ϵ-optimal solution $\bar{\mathbf{w}}_T=\sum_{t=1}^T\mathbf{w}_t/T$ in expectation with a complexity of

$$T=O\left(\frac{G_1L_2}{\epsilon},\frac{\sigma^2}{\epsilon^2},\frac{\Omega\sigma_0^2}{B\epsilon^2},\frac{\Omega\sigma_0^2}{n\epsilon^2}\right).$$

where Ω and $\mathbf{y}_T^*$ are defined similarly as in last theorem.

Why it matters

For smooth functions g_i, the iteration complexity is improved in the sense that the $O(1/\epsilon^2)$ dependence is scaled by the variance of the stochastic estimators. In contrast, for non-smooth g_i, the complexity always includes a term $\frac{G_1^2G_2^2}{\epsilon^2}$, regardless of the variance.

Proof. The proof is similar to that of previous theorem except that we use (5.39) instead of (5.46), and using (5.48) instead of using (5.49). Additionally, we use

$$\begin{aligned}\sum_{t=1}^T(\Gamma_{t+1}-\Gamma_t)&=\Gamma_{T+1}-\Gamma_1\le\frac{1}{n}\sum_{i=1}^n G_2\|\mathbf{w}_{T+1}-\mathbf{w}_T\|_2\|y_i-y_{i,T+1}\|_2 \qquad (5.61)\\ &\le\frac{G_2^2B}{2n\tau}\|\mathbf{w}_{T+1}-\mathbf{w}_T\|_2^2+\frac{\tau n/B}{n}D_\psi(\mathbf{y},\mathbf{y}_{T+1}).\end{aligned}$$

Combining this with (5.39), we have

$$\begin{aligned}&\mathbb{E}\left[\sum_{t=1}^T B_t(\mathbf{y})\right]\le\frac{G_2^2B}{2n\tau}\mathbb{E}[\|\mathbf{w}_{T+1}-\mathbf{w}_T\|_2^2]+\frac{\tau}{B}\mathbb{E}[D_\psi(\mathbf{y},\mathbf{y}_{T+1})] \qquad (5.62)\\ &+\frac{2}{n\lambda_2}\mathbb{E}[D_\psi(\mathbf{y},\tilde{\mathbf{y}}_1)]+\frac{(\lambda_3+\lambda_4)}{n}\sum_{t=1}^T\mathbb{E}\left[D_\psi(\bar{\mathbf{y}}_{t+1},\mathbf{y}_t)\right]+\frac{G_2^2}{2\lambda_3}\sum_{t=1}^T\mathbb{E}[\|\mathbf{w}_{t+1}-\mathbf{w}_t\|_2^2]\\ &+\frac{G_2^2}{2\lambda_4}\sum_{t=1}^T\mathbb{E}[\|\mathbf{w}_t-\mathbf{w}_{t-1}\|_2^2]+8\sigma_0^2\alpha T+\lambda_2\sigma_0^2T+\frac{\sigma_0^2\lambda_5}{2}T+\frac{1}{n\lambda_5}\mathbb{E}[D_\psi(\mathbf{y},\check{\mathbf{y}}_1)].\end{aligned}$$

Summing the inequalities in (5.32), (5.58), (5.62), and (5.49) over $t=1,\ldots,T$, we have

$$\mathbb{E}\Big[\sum_{t=1}^{T} F(\mathbf{w}_{t+1}, \mathbf{y}) - F(\mathbf{w}_*, \bar{\mathbf{y}}_{t+1})\Big] \leq \frac{1}{2\eta}\|\mathbf{w}_* - \mathbf{w}_1\|_2^2 - \frac{1}{2\eta}\sum_{t=1}^{T}\mathbb{E}[\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2]$$
$$+ \frac{\tau}{B}\mathbb{E}[D_\psi(\mathbf{y}, \mathbf{y}_1) - D_\psi(\mathbf{y}, \mathbf{y}_{T+1})] - \frac{\tau}{n}\sum_{t=1}^{T}\mathbb{E}\left[D_\psi(\bar{\mathbf{y}}_{t+1}, \mathbf{y}_t)\right]$$
$$+ \frac{\lambda_0\tau}{n}\mathbb{E}D_\psi(\mathbf{y}, \hat{\mathbf{y}}_1) + \frac{(n-B)\tau}{\lambda_0 nB}\sum_{t=1}^{T}\mathbb{E}\left[D_\psi(\bar{\mathbf{y}}_{t+1}, \mathbf{y}_t)\right],$$
$$+ \frac{G_2^2 B}{2n\tau}\mathbb{E}[\|\mathbf{w}_{T+1} - \mathbf{w}_T\|_2^2] + \frac{\tau}{B}\mathbb{E}[D_\psi(\mathbf{y}, \mathbf{y}_{T+1})]$$
$$+ \frac{2}{n\lambda_2}\mathbb{E}[D_\psi(\mathbf{y}, \tilde{\mathbf{y}}_1)] + \frac{(\lambda_3+\lambda_4)}{n}\sum_{t=1}^{T}\mathbb{E}\Big[D_\psi(\bar{\mathbf{y}}_{t+1}, \mathbf{y}_t)\Big] + \frac{G_2^2}{2\lambda_3}\sum_{t=1}^{T}\mathbb{E}[\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2]$$
$$+ \frac{G_2^2}{2\lambda_4}\sum_{t=1}^{T}\mathbb{E}[\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2^2] + 8\sigma_0^2\alpha T + \lambda_2\sigma_0^2 T + \frac{\sigma_0^2\lambda_5}{2}T + \frac{1}{n\lambda_5}\mathbb{E}[D_\psi(\mathbf{y}, \check{\mathbf{y}}_1)],$$
$$+ \frac{1}{4\eta}\sum_{t=1}^{T}\mathbb{E}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 + \eta T\sigma^2.$$

Similarly as before, if we let $\lambda_0 = \frac{2(n-B)}{B}$, $\frac{G_2^2}{2\lambda_3} = \frac{G_2^2}{2\lambda_4} = \frac{1}{16\eta}$, $\lambda_3 + \lambda_4 = 16\eta G_2^2 \leq \tau/2$, and $\frac{G_2^2 B}{2n\tau} \leq \frac{1}{8\eta}$, we observe that all the cumulated terms cancel out, leaving us the following:

$$\mathbb{E}\Big[\sum_{t=1}^{T} F(\mathbf{w}_{t+1}, \mathbf{y}) - F(\mathbf{w}_*, \bar{\mathbf{y}}_{t+1})\Big] \leq$$
$$\frac{1}{2\eta}\|\mathbf{w}_* - \mathbf{w}_1\|_2^2 + \left(\frac{\tau(1+\lambda_0 B/n)}{B} + \frac{2}{n\lambda_2} + \frac{1}{n\lambda_5}\right)\mathbb{E}D_\psi(\mathbf{y}, \mathbf{y}_1)$$
$$+ \eta T\sigma^2 + 8\sigma_0^2\alpha T + \lambda_2\sigma_0^2 T + \frac{\sigma_0^2\lambda_5}{2}T.$$

Let $\mathbf{y} = \mathbf{y}_T^* = \arg\max F(\bar{\mathbf{w}}_T, \mathbf{y})$. Since $\frac{1}{T}\sum_{t=1}^{T} F(\mathbf{w}_{t+1}, \mathbf{y}) \geq F(\bar{\mathbf{w}}_T, \mathbf{y}_T^*) = F(\bar{\mathbf{w}}_T)$ and $F(\mathbf{w}_*, \bar{\mathbf{y}}_{t+1}) \leq F(\mathbf{w}_*, \mathbf{y}_*)$, we have

$$\mathbb{E}\Big[F(\bar{\mathbf{w}}_T) - F(\mathbf{w}_*)\Big] \leq \frac{1}{2\eta T}\|\mathbf{w}_* - \mathbf{w}_1\|_2^2 + \frac{1}{T}\left(\frac{\tau(1+\lambda_0 B/n)}{B} + \frac{2}{n\lambda_2} + \frac{1}{n\lambda_5}\right)\Omega$$
$$+ \eta(\sigma^2) + 8\sigma_0^2\alpha + \lambda_2\sigma_0^2 + \frac{\sigma_0^2\lambda_5}{2}. \tag{5.63}$$

Let

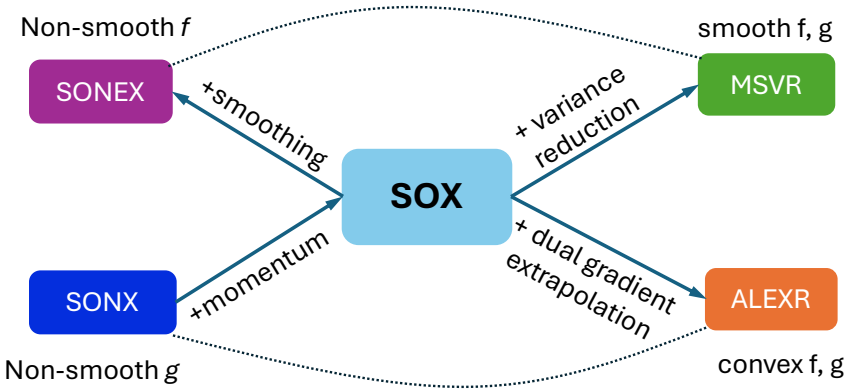


Fig. 5.1: Relationship between different algorithms for FCCO.

$$\alpha = \frac{\epsilon}{64\sigma_0^2}, \quad \lambda_2 = \frac{\epsilon}{8\sigma_0^2}, \quad \lambda_5 = \frac{\epsilon}{4\sigma_0^2}, \quad \eta = \min\left(\frac{\epsilon}{8\sigma^2}, \frac{1}{2G_1L_2}\right)$$
$$T \geq O\left(\max\left(\frac{\|\mathbf{w}_1 - \mathbf{w}_*\|_2^2}{32\eta\epsilon}, \frac{\Omega(1+\lambda_0 B/n)}{8B\epsilon\alpha}, \frac{\Omega}{4n\lambda_2\epsilon}, \frac{\Omega}{8n\lambda_5\epsilon}\right)\right).$$

Then the conditions $16\eta G_2^2 \leq \tau/2$, $\frac{G_2^2 B}{2n\tau} \leq \frac{1}{8\eta}$ hold for sufficiently small ϵ, and the RHS of (5.63) is less than ϵ. As a result, the complexity is in the order of

$$O\left(\max\left(\frac{G_1L_2}{\epsilon}, \frac{\sigma^2}{\epsilon^2}, \frac{\Omega\sigma_0^2}{B\epsilon^2}, \frac{\sigma_0^2\Omega}{n\epsilon^2}\right)\right).$$

□

Critical: The convergence results above remain valid for ALEXR-v2 even when the outer functions f_i are smooth. If f_i is a smooth Legendre function, ALEXR-v1 can also be applied and its convergence can be established. The key is to note that

$$\left\|\nabla\psi_i(\bar{y}_{i,t+1}) - \nabla\psi_i(y_{i,t})\right\|_2^2 = \left\|\nabla f_i^*(\bar{y}_{i,t+1}) - \nabla f_i^*(y_{i,t})\right\|_2^2 = \left\|\bar{\mathbf{u}}_{i,t} - \mathbf{u}_{i,t-1}\right\|_2^2,$$

where $\mathbf{u}_{i,t-1}$ is defined in Lemma 5.14 and $\bar{\mathbf{u}}_{i,t}$ is a virtual sequence similar to $\mathbf{u}_{i,t}$ (5.64) except that all coordinates are updated by:

$$\bar{\mathbf{u}}_{i,t} = \frac{1}{1+\alpha_t}\mathbf{u}_{i,t-1} + \frac{\alpha_t}{1+\alpha_t}\tilde{g}_{i,t}, \forall i. \tag{5.64}$$

Then, similar to the analysis of SOX, we can establish a bound of $\sum_{t=1}^{T}\sum_{i=1}^{n}\mathbb{E}[\left\|\bar{\mathbf{u}}_{i,t} - \mathbf{u}_{i,t-1}\right\|_2^2]$ and use it to prove the convergence of ALEXR-v1. However, it remains unclear whether ALEXR-v1 provides any convergence advantage over ALEXR-v2 when f_i are smooth.

5.4.5 Double-loop ALEXR for weakly convex inner functions

ALEXR can be also useful for solving non-convex FCCO with convex outer functions and weakly convex inner functions. In particular, we consider the following non-convex problem:

$$\min_{\mathbf{w}} \frac{1}{n}\sum_{i=1}^{n} f_i(g_i(\mathbf{w})) + r(\mathbf{w}),$$

where $g_i : \mathbb{R}^d \to \mathbb{R}^{d'}$ and $f_i : \mathbb{R}^{d'} \to \mathbb{R}$ satisfy the following conditions:

Assumption 5.12. *Assume*

(i) f_i is convex, G_1-Lipschitz continuous and $\partial f(g) \geq 0$.
(ii) each dimension of g_i is ρ_2-weakly convex and G_2-Lipschitz continuous.
(iii) $r(\mathbf{w})$ is a convex function.

The key idea is to solve the following quadratic problem sequentially:

$$\mathbf{w}_{t+1} \approx \arg\min \bar{F}(\mathbf{w}, \mathbf{w}_t) := \frac{1}{n}\sum_{i=1}^{n} f_i(g_i(\mathbf{w})) + \frac{\bar{\rho}}{2}\|\mathbf{w} - \mathbf{w}_t\|_2^2,$$

where $\bar{\rho} > \rho$, with ρ being the weak-convexity parameter of $F(\mathbf{w})$. We can employ ALEXR to solve $\min_{\mathbf{w}} \bar{F}(\mathbf{w}, \mathbf{w}_t)$ approximately up to an ϵ-level. This yields a double-loop scheme.

f_i	g_i	r	F	Algorithm	Convergence Measure	Complexity	Theorem
sm	-	0	ncx, sm	SOX	Stationary	$O\left(\frac{n\sigma_0^2}{B\epsilon^4}\right)$	Thm. 5.1
sm	mss	0	ncx	MSVR	Stationary	$O\left(\frac{n\sigma_0}{B\epsilon^3}\right)$	Thm. 5.2
sm	-	pm	ncx, sm	SOX	Stationary	$O\left(\frac{n\sigma_0^2}{B\epsilon^4}\right)$	Thm. 5.1
sm	mss	pm	ncx	MSVR	Stationary	$O\left(\frac{n\sigma_0}{B\epsilon^3}\right)$	Thm. 5.2
wc, nd	wc	0	ncx	SONX (v1)	Nearly Stationary	$O\left(\frac{n\sigma_0^2}{B\epsilon^8}\right)$	Thm. 5.3
sm, nd	wc	0	ncx	SONX (v1)	Nearly Stationary	$O\left(\frac{n\sigma_0^2}{B\epsilon^5}\right)$	Thm. 5.4
wc, nd	wc	0	ncx	SONX (v2)	Nearly Stationary	$O\left(\frac{n\sigma_0}{B\epsilon^6}\right)$	Thm. 5.5
sm, nd	wc	0	ncx	SONX (v2)	Nearly Stationary	$O\left(\frac{n\sigma_0}{B\epsilon^4}\right)$	Thm. 5.5
wc, pm	sm	0	ncx	SONEX (v1)	Approx. Stationary	$O\left(\frac{n\sigma_0^2}{B\epsilon^7}\right)$	Cor. 5.1
wc, pm	sm	0	ncx	SONEX (v2)	Approx. Stationary	$O\left(\frac{n\sigma_0}{B\epsilon^5}\right)$	Thm. 5.6
nd, cvx, f_i^* pm	sm, cvx	cvx, pm	cx	ALEXR (v2)	Obj. Gap	$O\left(\max\left(\frac{\sigma^2}{\epsilon^2}, \frac{n\sigma_0^2}{B\epsilon^2}\right)\right)$	Thm. 5.9
nd, cvx, f_i^* pm	cvx	cvx, pm	cx	ALEXR (v2)	Obj. Gap	$O\left(\max\left(\frac{1}{\epsilon^2}, \frac{n\sigma_0^2}{B\epsilon^2}\right)\right)$	Thm. 5.8
sm, nd, cvx	cvx	scx, pm	cx	ALEXR	Dist. Gap	$O\left(\max\left(\frac{1}{\mu\epsilon}, \frac{n\sigma_0^2}{B\epsilon}\right)\right)$	Thm.5.7
sm, nd, cvx	sm, cvx	scx, pm	cx	ALEXR	Dist. Gap	$O\left(\max\left(\frac{\sigma^2}{\mu\epsilon}, \frac{n\sigma_0^2}{B\epsilon}\right)\right)$	Thm.5.7
sm, nd, cvx, f_i^* pm	wc	cx, pm	ncx	ALEXR-DL	Nearly Stationary	$O\left(\max\left(\frac{1}{\epsilon^4}, \frac{n\sigma_0^2}{B\epsilon^4}\right)\right)$	-
nd, cvx, f_i^* pm	wc	cx, pm	ncx	ALEXR-DL	Approx. Stationary	$O\left(\max\left(\frac{1}{\epsilon^5}, \frac{n\sigma_0^2}{B\epsilon^5}\right)\right)$	-

Table 5.2: Complexity of solving FCCO $F(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^n f_i(g_i(\mathbf{w})) + r(\mathbf{w})$ under different conditions of f_i and g_i, where f_i is a deterministic Lipschitz continuous and g_i is mean Lipschitz continuous. pms means "proximal mapping is simple to compute", mss mean "mean squared smoothness", and ALEXR-DL denotes a double-loop method that employs ALEXR in the inner loop.

We highlight the key results as follows. If each f_i is non-smooth, the double loop method achieves a sample complexity of $O\left(\frac{n\sigma_0^2}{B\epsilon^6}\right)$ for finding a nearly ϵ-stationary solution. The analysis can be found in (Zhou et al., 2025).

If each f_i is L_1-smooth, the sample complexity improves to $O\left(\frac{nL_1\sigma_0^2}{B\epsilon^4}\right)$ for obtaining a nearly ϵ-stationary solution. This result further implies that, for non-smooth f_i, we may apply the Nesterov smoothing $\bar{f}_i$ in (5.20) with $\bar{\rho}_1 = 1/\epsilon$, so that $\bar{f}_i$ becomes $L_1 = \bar{\rho}_1$-smooth. Hence, Proposition 5.1 implies that the double-loop ALEXR algorithm can find an approximate ϵ-stationary stationary solution of $F(\mathbf{w})$ with a sample complexity $O\left(\frac{nL_1\sigma_0^2}{B\epsilon^4}\right) = O\left(\frac{n\sigma_0^2}{B\epsilon^5}\right)$. The analysis can be found in (Chen et al., 2025b).

Finally, we summarize the sample complexities of all methods introduced in this chapter in Table 5.2, and illustrate the relationship between different methods in Figure 5.1.

Algorithm 19 Abstract Stochastic Update Scheme for Convex FCCO

Initialize affine subspaces $\mathfrak{X}_0, \mathfrak{Y}_0, \mathfrak{g}_0, \mathfrak{G}_0$
for $t = 0, 1, \ldots, T-1$ **do**
  Sample a batch $\mathcal{B}_t \subset \{1, \ldots, n\}$, $|\mathcal{B}_t| = B$
  for each $i \in \mathcal{B}_t$ **do**
    Sample $\zeta_{i,t}, \tilde{\zeta}_{i,t}$ from $\mathbb{P}_i$
    $\mathfrak{g}_{t+1}^{(i)} = \mathfrak{g}_t^{(i)} + \operatorname{span}\{g_i(\hat{x}; \zeta_{i,t}) \mid \hat{x} \in \mathfrak{X}_t\}$

$$\mathfrak{Y}_{t+1}^{(i)} = \mathfrak{Y}_t^{(i)} + \operatorname{span}\left\{\arg\max_{y_i}\left\{y_i \hat{g}^{(i)} - f_i^*(y_i) - \frac{1}{\alpha} D_{\psi_i}(y_i, \hat{y}^{(i)})\right\} \mid \hat{g}^{(i)} \in \mathfrak{g}_{t+1}^{(i)}, \hat{y}^{(i)} \in \mathfrak{Y}_t^{(i)}\right\}$$

  end for
  For each $i \notin \mathcal{S}_t$, $\mathfrak{g}_{t+1}^{(i)} = \mathfrak{g}_t^{(i)}$, $\mathfrak{Y}_{t+1}^{(i)} = \mathfrak{Y}_t^{(i)}$
  $\mathfrak{G}_{t+1} = \mathfrak{G}_t + \operatorname{span}\left\{\frac{1}{B}\sum_{i \in \mathcal{B}_t} \hat{y}^{(i)} \nabla g_i(\hat{x}; \tilde{\zeta}_{i,t}) \mid \hat{x} \in \mathfrak{X}_t, \hat{y} \in \mathfrak{Y}_{t+1}\right\}$
  $\mathfrak{X}_{t+1} = \mathfrak{X}_t + \operatorname{span}\left\{\arg\min_x \left\{\hat{G}^\top x + r(x) + \frac{1}{2\eta}\|x - \hat{x}\|_2^2\right\} \mid \hat{x} \in \mathfrak{X}_t, \hat{G} \in \mathfrak{G}_{t+1}\right\}$
end for

5.4.6 Lower Bounds

In this section, we prove that the complexities of ALEXR for strongly convex and convex FCCO problems are nearly optimal by establishing the matching *lower* bounds.

What is a lower bound?

A lower bound states: for any algorithm in a certain class, there exists a "hard" optimization problem such that the algorithm cannot converge faster than a specified rate.

Lower bounds for convex optimization are typically derived under the standard oracle model, where the algorithm has access only to first-order information—either exact gradients in the deterministic setting or unbiased stochastic gradients in the stochastic setting. In the latter case, a classical result by Nemirovski and Yudin establishes that no stochastic algorithm using unbiased gradient oracles can achieve a convergence rate faster than $O(1/\sqrt{T})$ in terms of the objective gap after T iterations. For strongly convex problems, this lower bound improves to $O(1/T)$. Nevertheless, these lower bounds do not apply to convex FCCO problems or to ALEXR, because the algorithm does not have access to unbiased stochastic gradients.

Below, we establish lower bounds for an abstract stochastic update scheme described in Algorithm 19, where the symbol "+" denotes Minkowski addition. We consider an oracle model that, upon receiving a query point, returns unbiased stochastic function values and stochastic gradients of the inner functions g_i, as well as the solution to the proximal mirror-descent update of f_i^* with respect to a proximal function ψ_i. Since there are n inner functions in total, we assume that at each iteration the algorithm is allowed to access information from only B randomly selected in-

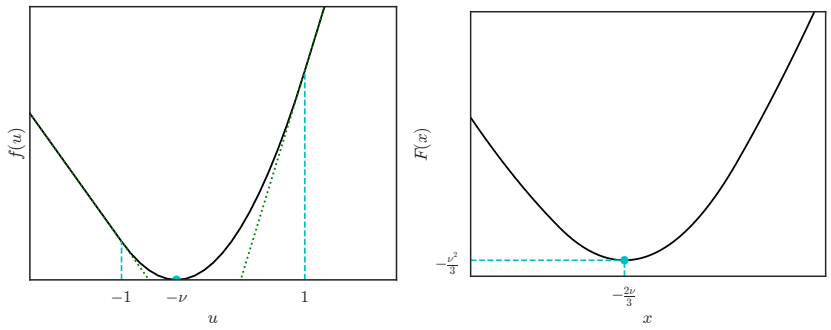


Fig. 5.2: Visualization of f (left) and F (right) in (5.65).

ner functions. Algorithm 19 is sufficiently general to encompass ALEXR, as well as SOX and MSVR.

Theorem 5.10 *Consider the abstract scheme (Algorithm 19) with an initialization* $\mathfrak{X}_0^{(i)} = \{0\}$, $\mathfrak{Y}_0^{(i)} = \{0\}$, $\mathfrak{g}_0^{(i)} = \emptyset$, $\mathfrak{G}_0^{(i)} = \emptyset$.

- *There exists a convex FCCO problem (5.26) with smooth f_i and μ-strongly convex r such that any algorithm in the abstract scheme requires at least $T = \Omega\left(\frac{n\sigma_0^2}{B\epsilon}\right)$ iterations to find an $\bar{x}$ that satisfies $\mathbb{E}\left[\frac{\mu}{2}\|\bar{x} - x_*\|_2^2\right] \leq \epsilon$ or $\mathbb{E}[F(\bar{x}) - F(x_*)] \leq \epsilon$.*
- *There exists a convex FCCO problem (5.26) with non-smooth f_i such that any algorithm in the abstract scheme requires at least $T = \Omega\left(\frac{n\sigma_0^2}{B\epsilon^2}\right)$ iterations to find an $\bar{x}$ that satisfies $\mathbb{E}[F(\bar{x}) - F(x_*)] \leq \epsilon$.*

Why it matters

In light of this theorem, we see that ALEXR (v1/v2) attains a nearly optimal complexity up to a logarithmic factor for solving strongly convex FCCO problems, as its upper bounds are $\tilde{O}\left(\max\left(\frac{1}{\mu\epsilon}, \frac{n\sigma_0^2}{B\epsilon}\right)\right)$. Moreover, ALEXR-v2 achieves the optimal complexity for solving convex FCCO problems with non-smooth outer functions.

Proof. We construct the hard problems for (i) smooth f_i; and (ii) non-smooth f_i separately.

(i) Smooth f_i and strongly convex r: Consider the following strongly convex FCCO problem

$$\min_{x\in\mathcal{X}} \quad F(x) = \frac{1}{n}\sum_{i=1}^{n} f(g_i(x)) + r(x),$$
$$f(u) = \begin{cases} (\nu-1)u + \frac{1}{2}(\nu-1)^2 + \nu - 1 - \frac{\nu^2}{2}, & u \in (-\infty, -1) \\ \frac{1}{2}(u+\nu)^2 - \frac{\nu^2}{2}, & u \in [-1, 1] \\ (1+\nu)u + \frac{1}{2}(1+\nu)^2 - 1 - \nu - \frac{\nu^2}{2}, & u \in (1, \infty) \end{cases}, \quad r(x) = \frac{1}{4n}\|x\|_2^2, \tag{5.65}$$

where $\mathcal{X} = [-1, 1]^n$, the outer function $f : \mathbb{R} \to \mathbb{R}$ is smooth and Lipschitz continuous for some $\nu \in (0, 1/2)$. Besides, the inner function $g_i : \mathbb{R}^n \to \mathbb{R}$ is $g_i(x) = \mathbb{E}_{\zeta\sim\mathbb{P}}[g_i(x;\zeta)]$ and $g_i(x;\zeta) = x_i + \zeta$, where ζ follows a distribution $\mathbb{P}$ defined below:

$$\mathbb{P} : \begin{cases} \Pr(\zeta = -\nu) = 1 - p, \\ \Pr(\zeta = \nu(1-p)/p) = p \end{cases}, \text{ where } p \coloneqq \frac{\nu^2}{\sigma_0^2} < 1.$$

We will determine the values of ν later. We can verify that

$$\mathbb{E}_\zeta[|g_i(x;\zeta) - g_i(x)|^2] = \mathbb{E}_\zeta[\zeta^2] = \nu^2(1-p) + \frac{\nu^2(1-p)^2}{p} = \frac{\nu^2(1-p)}{p} \le \sigma_0^2.$$

By the definition of convex conjugate, for any $y_i \in \mathbb{R}$ we have

$$\begin{aligned} f^*(y_i) = \max\Bigg\{ & \sup_{u<-1}\left\{uy_i - \left((\nu-1)u + \frac{1}{2}(\nu-1)^2 + \nu - 1 - \frac{\nu^2}{2}\right)\right\}, \\ & \sup_{-1\le u\le 1}\left\{uy_i - \frac{1}{2}(u+\nu)^2 + \frac{\nu^2}{2}\right\}, \\ & \sup_{u>1}\left\{uy_i - \left((1+\nu)u + \frac{1}{2}(1+\nu)^2 - 1 - \nu - \frac{\nu^2}{2}\right)\right\}\Bigg\} \end{aligned} \tag{5.66}$$

$$= \begin{cases} +\infty, & y_i \in (-\infty, \nu-1) \cup (\nu+1, \infty) \\ \frac{1}{2}(y_i - \nu)^2, & y_i \in [\nu-1, \nu+1]. \end{cases} \tag{5.67}$$

We define $F_i(x_i) \coloneqq f(g_i(x)) + \frac{1}{4}[x_i]^2$ such that $F(x) = \frac{1}{n}\sum_{i=1}^n F_i(x_i)$. Let $x_* = \arg\min_{x\in\mathcal{X}} F(x)$. Since the problem is separable over the coordinates, we have $x_{i,*} = \arg\min_{x\in[-1,1]} F_i(x_i)$. Thus, we have $x_{i,*} = -\frac{2\nu}{3}$ and $F_i(x_{i,*}) = -\frac{\nu^2}{3}$.

Since $\mathbb{P}_i = \mathbb{P}$ in the "hard" problem (5.65), the abstract scheme (Algorithm 19) only needs to sample shared $\zeta_t, \tilde{\zeta}_t \sim \mathbb{P}$ for all coordinates $i \in \mathcal{S}_t$ in the t-th iteration. For any $i \in [n]$, suppose that $\mathfrak{g}_\tau^{(i)} = \emptyset$ or $\{-\nu\}$, $\mathfrak{y}_\tau^{(i)} = \{0\}$, $\mathfrak{x}_\tau^{(i)} = \{0\}$ for all $\tau \le t$. Note that when $\mathfrak{g}_\tau^{(i)} = \emptyset$, it means that the corresponding $y^{(i)}$ will not be updated. Then,

- If $i \notin \mathcal{B}_t$, the abstract scheme (Algorithm 19) leads to

$$\mathfrak{g}_{t+1}^{(i)} = \emptyset \text{ or } \{-\nu\}, \quad \mathfrak{y}_{t+1}^{(i)} = \{0\}, \quad \mathfrak{x}_{t+1}^{(i)} = \{0\}.$$

- If $i \in \mathcal{B}_t$ and $\zeta_t = -\nu$, the abstract scheme (Algorithm 19) proceeds as

$$\begin{aligned}
\mathfrak{g}_{t+1}^{(i)} &= \mathfrak{g}_t^{(i)} + \operatorname{span}\left\{\hat{x}_i + \zeta_t \mid \hat{x}_i \in \mathfrak{X}_t^{(i)}\right\}, \\
\mathfrak{Y}_{t+1}^{(i)} &= \mathfrak{Y}_t^{(i)} \\
&+ \operatorname{span}\left\{\operatorname*{arg\,max}_{y_i \in [\nu-1, \nu+1]} \left\{y_i \hat{g}_i - \frac{1}{2}(y_i - \nu)^2 - \frac{1}{\alpha} D_{\psi_i}(y_i, \hat{y}_i)\right\} \mid \hat{g}_i \in \mathfrak{g}_{t+1}^{(i)}, \hat{y}_i \in \mathfrak{Y}_t^{(i)}\right\}, \\
\mathfrak{X}_{t+1}^{(i)} &= \mathfrak{X}_t^{(i)} \\
&+ \operatorname{span}\left\{\operatorname*{arg\,min}_{x_i \in [-1,1]} \left\{\frac{1}{B}\hat{y}_i x_i + \frac{1}{4n}[x_i]^2 + \frac{1}{2\eta}(x_i - \hat{x}_i)^2\right\} \mid \hat{y}_i \in \mathfrak{Y}_{t+1}^{(i)}, \hat{x}_i \in \mathfrak{X}_t^{(i)}\right\}.
\end{aligned}$$

Then, we can derive that $\mathfrak{g}_{t+1}^{(i)} = \emptyset$ or $\{-\nu\}$, $\mathfrak{Y}_{t+1}^{(i)} = \{0\}$, and $\mathfrak{X}_{t+1}^{(i)} = \{0\}$.

To sum up, given the event $\bigcap_{\tau=1}^{t}\{\mathfrak{g}_\tau^{(i)} = \emptyset \text{ or } \{-\nu\}, \mathfrak{Y}_\tau^{(i)} = \{0\}, \mathfrak{X}_\tau^{(i)} = \{0\}\}$, we can make sure that $\{\mathfrak{g}_{t+1}^{(i)} = \emptyset \text{ or } \{-\nu\} \wedge \mathfrak{Y}_{t+1}^{(i)} = \{0\} \wedge \mathfrak{X}_{t+1}^{(i)} = \{0\}\}$ for the abstract scheme in Algorithm 19 when one of the following mutually exclusive events happens:

- Event I: $i \notin \mathcal{B}_t$;
- Event II: $i \in \mathcal{B}_t$ and $\zeta_t = -\nu$.

Note that the random variable ζ_t is independent of $\mathcal{B}_t$. Thus, the probability of the event $E_{t+1}^{(i)} \coloneqq \{\mathfrak{g}_{t+1}^{(i)} = \emptyset \text{ or } \{-\nu\} \wedge \mathfrak{Y}_{t+1}^{(i)} = \{0\} \wedge \mathfrak{X}_{t+1}^{(i)} = \{0\}\}$ conditioned on $\bigcap_{\tau=1}^{t} E_\tau^{(i)}$ can be bounded as

$$\begin{aligned}
\Pr\left[E_{t+1}^{(i)} \mid \bigcap_{\tau=1}^{t} E_\tau^{(i)}\right] &= \mathbb{P}\left[\left\{\mathfrak{g}_{t+1}^{(i)} = \emptyset \text{ or } \{-\nu\} \wedge \mathfrak{Y}_{t+1}^{(i)} = \{0\} \wedge \mathfrak{X}_{t+1}^{(i)} = \{0\}\right\} \mid \bigcap_{\tau=1}^{t} E_\tau^{(i)}\right] \\
&\geq \mathbb{P}\left[\{i \notin \mathcal{B}_t\}\right] + \mathbb{P}\left[\{\{i \in \mathcal{B}_t\} \wedge \{\zeta_t = -\nu\}\}\right] \\
&= \mathbb{P}\left[\{i \notin \mathcal{B}_t\}\right] + \mathbb{P}\left[\{i \in \mathcal{B}_t\}\right]\mathbb{P}\left[\{\zeta_t = -\nu\}\right] \\
&= \left(1 - \frac{B}{n}\right) + \frac{B}{n}(1-p) = 1 - \frac{Bp}{n}.
\end{aligned}$$

Since $\mathcal{B}_t$ and ζ_t in different iterations t are mutually independent, we have

$$\Pr\left[E_T^{(i)}\right] \geq \mathbb{P}\left[\bigcap_{t=0}^{T-1} E_{t+1}^{(i)}\right] = \prod_{t=0}^{T-1} \mathbb{P}\left[E_{t+1}^{(i)} \mid \bigcap_{t=1}^{t} E_t^{(i)}\right] = \left(1 - \frac{Bp}{n}\right)^T > 3/4 - \frac{TBp}{n},$$

where the last inequality is due to the Bernoulli inequality $(1+x)^r \geq 1 + rx$ for every integer $r \geq 1$ and $x \geq -1$.

Thus, if $T < \frac{n}{4Bp}$ we have $\Pr\left[E_T^{(i)}\right] > \frac{1}{2}$. Let us set $\nu = 3\sqrt{2\epsilon}$ such that $p = \frac{\nu^2}{\sigma_0^2} = \frac{18\epsilon}{\sigma_0^2}$. For any $i \in [n]$ and any output $\bar{x}_i \in \mathfrak{X}_T^{(i)}$, we have

$$\begin{aligned}\mathbb{E}\left[\left(\bar{x}_i - x_{i,*}\right)^2\right] &= \mathbb{E}\left[\mathbb{I}_{E_T^{(i)}}\left(\bar{x}_i - x_{i,*}\right)^2 + \mathbb{I}_{\overline{E_T^{(i)}}}\left(\bar{x}^{(i)} - x_{i,*}\right)^2\right] \\ &\geq \mathbb{E}\left[\mathbb{I}_{E_T^{(i)}}\left(\bar{x}_i - x_{i,*}\right)^2\right] \\ &= \mathbb{E}\left[\mathbb{I}_{E_T^{(i)}}\left(x_{i,*}\right)^2\right] = \Pr\left[E_T^{(i)}\right]\left(x_{i,*}\right)^2 > \frac{2\nu^2}{9} = 4\epsilon,\end{aligned}$$

where $\mathbb{I}_E$ denotes the indicator function of an event E. Moreover, we have

$$\begin{aligned}\mathbb{E}[F_i(\bar{x}_i) - F_i(x_{i,*})] &= \mathbb{E}\left[\mathbb{I}_{E_T^{(i)}}\left(F_i(\bar{x}_i) - F_i(x_{i,*})\right) + \mathbb{I}_{\overline{E_T^{(i)}}}\left(F_i(\bar{x}_i) - F_i(x_{i,*})\right)\right] \\ &\geq \mathbb{E}\left[\mathbb{I}_{E_T^{(i)}}\left(F_i(\bar{x}_i) - F_i(x_{i,*})\right)\right] \\ &= \mathbb{E}\left[\mathbb{I}_{E_T^{(i)}}\left(F_i(0) - F_i(x_{i,*})\right)\right] = \Pr[E_T^{(i)}]\left(F_i(0) - F_i(x_{i,*})\right) \\ &> \frac{\nu^2}{6} > \epsilon.\end{aligned}$$

Since the derivations above hold for arbitrary $i \in [n]$ and the $r(x)$ in (5.65) is $\frac{1}{2n}$-strongly convex ($\mu = \frac{1}{2n}$), we can derive that

$$\mathbb{E}\left[\frac{\mu}{2}\left\|\bar{x} - x_*\right\|_2^2\right] = \mathbb{E}\left[\frac{1}{4n}\left\|\bar{x} - x_*\right\|_2^2\right] = \frac{1}{4n}\sum_{i=1}^{n}\mathbb{E}\left[\left(\bar{x}_i - x_{i,*}\right)^2\right] > \epsilon,$$

$$\mathbb{E}\left[F(\bar{x}) - F(x_*)\right] = \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\left[F_i(\bar{x}_i) - F_i(x_{i,*})\right] > \epsilon.$$

Thus, to find an output $\bar{x}$ that satisfies $\mathbb{E}\left[\frac{\mu}{2}\left\|\bar{x} - x_*\right\|_2^2\right] \leq \epsilon$ or $\mathbb{E}\left[F(\bar{x}) - F(x_*)\right] \leq \epsilon$, the abstract scheme requires at least $T \geq \frac{n}{4Bp} = \frac{n\sigma_0^2}{72B\epsilon}$ iterations.

(ii) Non-smooth f_i: Let $g_i(x) = \mathbb{E}_\zeta[x_i + \zeta] = x_i$ be defined the same as in the smooth case. Let $F_i(x_i) \coloneqq f(g_i(x)) + \frac{\alpha}{2}[x_i]^2 = \beta\max\{x_i, -\nu\} + \frac{\alpha}{2}[x_i]^2$ such that $F(x) = \frac{1}{n}\sum_{i=1}^{n}F_i(x_i)$, where $\alpha, \beta > 0$. Let the domain $\mathcal{X}$ be $[-2\nu, 2\nu]^n$. Hence, f is β-Lipschitz continuous and F is α-strongly convex. By the definition of convex conjugate, we have $f(\hat{g}_i) = \max_{y_i \in [0,\beta]}\left\{y_i\hat{g}_i - \nu(\beta - y_i)\right\}$.

Since the problem is separable over the coordinates, we have

$$x_{i,*} = \operatorname*{arg\,min}_{x \in [-2\nu, 2\nu]} F_i(x_i) = \operatorname*{arg\,min}_{x_i \in [-2\nu, 2\nu]}\left\{\beta\max\{x_i, -\nu\} + \frac{\alpha}{2}[x_i]^2\right\}.$$

Considering

$$F_i(x_i) = \begin{cases}\beta x_i + \frac{\alpha}{2}[x_i]^2 & x_i \geq -\nu \\ -\beta\nu + \frac{\alpha}{2}[x_i]^2 & x_i < -\nu\end{cases},$$

we have

$$x_{i,*} = \begin{cases} -\beta/\alpha & \text{if } \alpha > \beta/\nu \\ -\nu & \text{if } \alpha \in \frac{\beta}{\nu}[0,1] \end{cases}, \quad F_i(x_{i,*}) \le \begin{cases} -\beta^2/(2\alpha) & \text{if } \alpha > \beta/\nu \\ -\beta\nu/2 & \text{if } \alpha \in \frac{\beta}{\nu}[0,1]. \end{cases}$$

Since $F_i(0) = 0$, we can derive that $F_i(0) - F_i(x_{i,*}) \ge \frac{1}{2}\min\{\beta\nu, \beta^2/\alpha\}$. Consider an arbitrary $i \in [n]$. Suppose that $\mathfrak{g}_\tau^{(i)} = \emptyset$ or $\{-\nu\}$, $\mathfrak{X}_\tau^{(i)} = \{0\}$, $\mathfrak{Y}_\tau^{(i)} = \{0\}$ for all $\tau \le t$.

- If $i \notin \mathcal{B}_t$, the abstract scheme (Algorithm 19) leads to

$$\mathfrak{g}_{t+1}^{(i)} = \emptyset \text{ or } \{-\nu\}, \quad \mathfrak{Y}_{t+1}^{(i)} = \{0\}, \quad \mathfrak{X}_{t+1}^{(i)} = \{0\}.$$

- If $i \in \mathcal{B}_t$, the abstract scheme (Algorithm 19) proceeds as

$$\begin{aligned}
&\mathfrak{g}_{t+1}^{(i)} = \mathfrak{g}_t^{(i)} + \operatorname{span}\left\{\hat{x}_i + \zeta_t \mid \hat{x}_i \in \mathfrak{X}_t^{(i)}\right\}, \\
&\mathfrak{Y}_{t+1}^{(i)} = \mathfrak{Y}_t^{(i)} \\
&+ \operatorname{span}\left\{\operatorname*{arg\,max}_{y_i \in [0,\beta]} \left\{y_i \hat{g}_i - \nu(\beta - y_i) - \frac{1}{\alpha} D_\psi(y_i, \hat{y}_i)\right\} \mid \hat{g}_i \in \mathfrak{g}_{t+1}^{(i)}, \hat{y}_i \in \mathfrak{Y}_t^{(i)}\right\}, \\
&\mathfrak{X}_{t+1}^{(i)} = \mathfrak{X}_t^{(i)} \\
&+ \operatorname{span}\left\{\operatorname*{arg\,min}_{x_i \in [-2\nu, 2\nu]} \left\{\frac{1}{B}\hat{y}_i x_i + \frac{1}{n}[x_i]^2 + \frac{1}{2\eta}(x_i - \hat{x}_i)^2\right\} \mid \hat{y}_i \in \mathfrak{Y}_{t+1}^{(i)}, \hat{x}_i \in \mathfrak{X}_t^{(i)}\right\}.
\end{aligned}$$

Due to the same reason as in the smooth f_i case, the probability of the event $E_T^{(i)} \coloneqq \{\mathfrak{g}_T^{(i)} = \emptyset \text{ or } \{-\nu\} \wedge \mathfrak{Y}_T^{(i)} = \{0\} \wedge \mathfrak{X}_T^{(i)} = \{0\}\}$ can be bounded as

$$\Pr\left[E_T^{(i)}\right] \ge \mathbb{P}\left[\bigcap_{t=0}^{T-1} E_{t+1}^{(i)}\right] = \prod_{t=0}^{T-1} \mathbb{P}\left[E_{t+1}^{(i)} \mid \bigcap_{t=1}^{t} E_t^{(i)}\right] = \left(1 - \frac{Bp}{n}\right)^T > 3/4 - \frac{TBp}{n}.$$

Thus, if $T < \frac{n}{4Bp}$ we have $\mathbb{P}\left[E_T^{(i)}\right] > \frac{1}{2}$. Let us set $\beta = G_1$, $\nu = \frac{4\epsilon}{G_1}$ such that $p \coloneqq \frac{\nu^2}{\sigma_0^2} = \frac{16\epsilon^2}{G_1^2\sigma_0^2}$. For any $i \in [n]$ and any output $\bar{x}_i \in \mathfrak{X}_T^{(i)}$, we have

$$\begin{aligned}
\mathbb{E}[F_i(\bar{x}_i) - F_i(x_{i,*})] &= \mathbb{E}\left[\mathbb{I}_{E_T^{(i)}}\left(F_i(\bar{x}_i) - F_i(x_{i,*})\right) + \mathbb{I}_{\overline{E_T^{(i)}}}\left(F_i(\bar{x}_i) - F_i(x_{i,*})\right)\right] \\
&\ge \mathbb{E}\left[\mathbb{I}_{E_T^{(i)}}\left(F_i(\bar{x}_i) - F_i(x_{i,*})\right)\right] \\
&= \mathbb{E}\left[\mathbb{I}_{E_T^{(i)}}\left(F_i(0) - F_i(x_{i,*})\right)\right] \\
&= \Pr[E_T^{(i)}]\left(F_i(0) - F_i(x_{i,*})\right) > \min\{\beta\nu, \beta^2/\alpha\}/4 = \epsilon.
\end{aligned}$$

Since the derivations above hold for arbitrary $i \in [n]$, we can derive that

$$\mathbb{E}[F(\bar{x}) - F(x_*)] = \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}[F_i(\bar{x}_i) - F_i(x_{i,*})] > \epsilon.$$

Thus, to find an output $\bar{x}$ that satisfies $\mathbb{E}[F(\bar{x}) - F(x_*)] \leq \epsilon$, the abstract scheme requires at least $T \geq \frac{n}{4Bp} = \frac{nG_1^2\sigma_0^2}{64B\epsilon^2}$ iterations. □

Critical: From the proof of the non-smooth case, we can see that even when the overall objective is strongly convex, the lower bound complexity is still $T = \Omega\left(\frac{n\sigma_0^2}{B\epsilon^2}\right)$ as long as f_i is non-smooth. This behavior contrasts with standard strongly stochastic optimization with an optimal complexity of $O(1/\epsilon)$ and highlights a fundamental challenge in solving compositional problems.

5.5 Stochastic Optimization of Compositional OCE

The goal of this section is to present and analyze stochastic algorithms for solving compositional OCE (COCE) risk minimization as introduced in Chapter 3. In particular, we consider the following abstract problem:

$$\min_{\mathbf{w}\in\mathbb{R}^d, \boldsymbol{\nu}\in\mathbb{R}^n} F(\mathbf{w}, \boldsymbol{\nu}) := \frac{1}{n}\sum_{i=1}^{n} F_i(\mathbf{w}, \nu_i), \tag{5.68}$$

where

$$F_i(\mathbf{w}, \nu_i) = \mathbb{E}_{\zeta\sim\mathbb{P}_i}[\Phi_i(\mathbf{w}, \nu_i; \zeta)], \quad \Phi_i(\mathbf{w}, \nu_i; \zeta) = \tau\phi^*\left(\frac{s_i(\mathbf{w}; \zeta) - \nu_i}{\tau}\right) + \nu_i,$$

where $\tau > 0$ is a constant.

In the special case when $\phi^*(\cdot) = [\cdot]_+/\alpha$ for some $\alpha \in (0, 1)$, the general COCE minimization problem reduces to

$$\min_{\mathbf{w},\boldsymbol{\nu}} F(\mathbf{w}, \boldsymbol{\nu}) := \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}_{\zeta\sim\mathbb{P}_i} \frac{[s_i(\mathbf{w}; \zeta) - \nu_i]_+}{\alpha} + \nu_i. \tag{5.69}$$

We refer to this problem as the **compositional CVaR minimization (CCVaR)** problem. The direct one-way partial AUC optimization problem (2.39) can be reformulated as an instance of CCVaR minimization as shown in (6.26).

In the special case when $\phi^*(\cdot) = \exp(\cdot) - 1$, the problem (5.68) reduces to

$$\min_{\mathbf{w}} F(\mathbf{w}) := \frac{1}{n}\sum_{i=1}^{n} \tau \log\left(\mathbb{E}_{\zeta\sim\mathbb{P}_i} \exp\left(\frac{s_i(\mathbf{w}; \zeta)}{\tau}\right)\right). \tag{5.70}$$

Algorithm 20 The ASGD Algorithm for solving (5.68)

Initialize $\mathbf{w}_0, \boldsymbol{\nu}_0$, step sizes η_t and γ_t
for $t = 0, 1 \ldots, T-1$ **do**
  Sample $\mathcal{B}_t \subset \{1, \ldots, n\}$ and $|\mathcal{B}_t| = B$
  for each $i \in \mathcal{B}_t$ **do**
    Update $\boldsymbol{\nu}_{i,t+1} = \boldsymbol{\nu}_{i,t} - \gamma_t \partial_2 \Phi_i(\mathbf{w}_t, \boldsymbol{\nu}_{i,t}; \zeta_{i,t})$
  end for
  Compute $\mathbf{z}_t = \frac{1}{B} \sum_{i \in \mathcal{B}_t} \partial_1 \Phi_i(\mathbf{w}_t, \boldsymbol{\nu}_{i,t}; \zeta_{i,t})$
  Update $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{z}_t$
end for

We refer to this problem as the **compositional entropic risk minimization (CERM)** problem. The cross-entropy loss for multi-class classification, the listwise cross-entropy loss for ranking, the indirect one-way partial AUC loss for imbalanced classification, and the contrastive losses for representation learning discussed in Chapter 2 are all instances of the CERM problem. In particular, for cross-entropy loss minimization, the proposed framework becomes especially relevant when the number of classes is very large, so that the normalization term in the loss cannot be computed efficiently. This setting naturally motivates the stochastic algorithms developed in this section.

Although we can cast the CERM problem into a special instance of FCCO, there remain some gaps to be filled. (i) For the convex CERM problem with a convex loss function $s_i(\cdot; \zeta)$, the ALEXR algorithm and its convergence analysis are not directly applicable, since the outer function $f(\cdot) = \tau \log(\cdot)$ is *not* convex, as required by ALEXR. Consequently, a convergence rate of $O(1/\epsilon^2)$ for solving convex CERM remains to be developed. (ii) For the CCVaR problem, the optimal solution of $\boldsymbol{\nu}$ given $\mathbf{w}$ is typically difficult to derive in closed form, and hence the problem cannot be reduced to an instance of FCCO. As a result, previous analyses for FCCO do not directly apply. We address these gaps in this section.

5.5.1 A Basic Algorithm

For optimizing the general COCE minimization problem, we present a basic stochastic algorithm in Algorithm 20. It alternates the stochastic block-coordinate update for $\boldsymbol{\nu}$ and a SGD update for $\mathbf{w}$, which is referred to as ASGD. Below, we present its convergence analysis for both convex and non-convex loss functions.

5.5.1.1 Convex loss

For notational simplicity, we set $\tau = 1$ throughout the analysis.

Assumption 5.13. *$s_i(\cdot, \zeta)$ is a convex function.*

Lemma 5.20 *$F(\mathbf{w}, \boldsymbol{\nu})$ is jointly convex in terms of $(\mathbf{w}^\top, \boldsymbol{\nu}^\top)^\top$ if $s_i(\cdot; \zeta)$ is convex.*

Proof. We prove that $\Phi_i(\mathbf{w}, \nu_i; \zeta)$ is jointly convex in terms of $(\mathbf{w}^\top, \nu_i)^\top$. Then the convexity of $F(\mathbf{w}, \boldsymbol{\nu})$ follows. Let $\mathbf{u} = (\mathbf{w}^\top, \nu)^\top$. Consider $\mathbf{u}_1, \mathbf{u}_2$, $\alpha \in [0, 1]$, and $\bar{\mathbf{u}} = \alpha\mathbf{u}_1 + (1-\alpha)\mathbf{u}_2$. Then

$$\Phi_i(\bar{\mathbf{u}}; \zeta) = \phi^*(s_i(\bar{\mathbf{w}}; \zeta) - \bar{\nu}) + \bar{\nu}.$$

If $s_i(\cdot; \zeta)$ is convex, we have $s_i(\bar{\mathbf{w}}; \zeta) \leq_i (\mathbf{w}_1; \zeta) + (1-\alpha)s_i(\mathbf{w}_2; \zeta)$. Since $\phi^*(\cdot)$ is non-decreasing (cf. Lemma 2.3), we have

$$\phi^*(s_i(\bar{\mathbf{w}}; \zeta) - \bar{\nu}) \leq \phi^*(\alpha(s_i(\mathbf{w}_1; \zeta) - \nu_1) + (1-\alpha)(s_i(\mathbf{w}_2; \zeta) - \nu_2)).$$

Since $\phi^*(\cdot)$ is convex, we further have

$$\begin{aligned}&\phi^*(\alpha(s_i(\mathbf{w}_1; \zeta) - \nu_1) + (1-\alpha)(s_i(\mathbf{w}_2; \zeta) - \nu_2))\\ &\leq \alpha\phi^*(s_i(\mathbf{w}_1; \zeta) - \nu_1) + (1-\alpha)\phi^*(s_i(\mathbf{w}_2; \zeta) - \nu_2).\end{aligned}$$

As a result,

$$\Phi_i(\bar{\mathbf{u}}; \zeta) \leq \alpha\Phi_i(\mathbf{u}_1; \zeta) + (1-\alpha)\Phi_i(\mathbf{u}_2; \zeta),$$

which proves the convexity of $\Phi_i(\mathbf{u}; \zeta)$.

□

Assumption 5.14. *Assume that either of the following conditions hold:*

- *(i) $F(\mathbf{w}, \boldsymbol{\nu})$ is smooth satisfying:*

$$\|\nabla_1 F(\mathbf{w}, \boldsymbol{\nu})\|_2^2 + \|\nabla_2 F(\mathbf{w}, \boldsymbol{\nu})\|_2^2 \leq 2L_F(F(\mathbf{w}, \boldsymbol{\nu}) - F(\mathbf{w}_*, \boldsymbol{\nu}_*)),$$

- *(ii) $F(\mathbf{w}, \boldsymbol{\nu})$ non-smooth such that for any $\mathbf{v}_1 \in \partial_1 F(\mathbf{w}, \boldsymbol{\nu}), \mathbf{v}_{2,i} \in \partial_2 F_i(\mathbf{w}, \nu_i)$ it holds*

$$\|\mathbf{v}_1\|_2^2 \leq G_1^2, \quad |\mathbf{v}_{2,i}|^2 \leq G_2^2,$$

where $\mathbf{w}_*, \boldsymbol{\nu}_*$ denotes an optimal solution to (5.68), and $\nabla_1 F(\mathbf{w}, \boldsymbol{\nu})(\partial_1 F(\mathbf{w}, \boldsymbol{\nu}))$, and $\nabla_2 F(\mathbf{w}, \boldsymbol{\nu})(\partial_2 F(\mathbf{w}, \boldsymbol{\nu}))$ denote (partial) gradients with respect to $\mathbf{w}, \boldsymbol{\nu}$, respectively.

Critical: For CERM, the smoothness assumption is satisfied when $s_i(\mathbf{w}; \zeta)$ is bounded, Lipschitz, and smooth. For CCVaR, the non-smoothness assumption is satisfied when $s_i(\mathbf{w}; \zeta)$ is bounded and Lipschitz.

Assumption 5.15 (Bounded Variance)**.** *There exist $\sigma_1^2, \sigma_2^2, \delta^2$ such that*

$$\mathbb{E}_\zeta \|\nabla_1 \Phi_i(\mathbf{w}, \nu_i; \zeta) - \nabla_1 F_i(\mathbf{w}, \nu_i)\|_2^2 \le \sigma_1^2, \quad \forall i \in [n],$$
$$\mathbb{E}_\zeta \|\nabla_2 \Phi_i(\mathbf{w}, \nu_i; \zeta) - \nabla_2 F_i(\mathbf{w}, \nu_i)\|_2^2 \le \sigma_2^2, \quad \forall i \in [n],$$
$$\frac{1}{n}\sum_{i=1}^n \|\nabla_1 F_i(\mathbf{w}, \nu_i) - \nabla_1 F(\mathbf{w}, \boldsymbol{\nu})\|_2^2 \le \delta^2.$$

In the non-smooth case, the gradients above are replaced by subgradients. The subsequent analysis proceeds analogously.

Lemma 5.21 *Let* $D_{\mathbf{w},0}^2 \coloneqq \mathbb{E}\|\mathbf{w}_0 - \mathbf{w}_*\|_2^2$ *and* $\eta_t = \eta$*, we have*

$$\frac{1}{T}\sum_{t=0}^{T-1}(2\mathbb{E}[\nabla_1 F(\mathbf{w}_t, \boldsymbol{\nu}_t)^\top(\mathbf{w}_t - \mathbf{w}_*)] - \eta\mathbb{E}\|\nabla_1 F(\mathbf{w}_t, \boldsymbol{\nu}_t)\|_2^2) \le \frac{D_{\mathbf{w},0}^2}{\eta T} + \eta\sigma^2.$$

where $\boldsymbol{\nu}_t = (\nu_{1,t}, \ldots, \nu_{n,t})^\top$ *and* $\sigma^2 = \frac{\sigma_1^2}{B} + \frac{\delta^2(n-B)}{B(n-1)}$.

Proof. Let $\mathbb{E}_t$ denote the expectation over the random samples in the t-th iteration. First, we note that $\mathbb{E}_t[\mathbf{z}_t] = \nabla_1 F(\mathbf{w}_t, \boldsymbol{\nu}_t)$. Similar to Lemma 5.2, we have

$$\begin{aligned}
&\mathbb{E}_t\|\mathbf{z}_t - \nabla_1 F(\mathbf{w}_t, \boldsymbol{\nu}_t)\|_2^2 \\
&= \mathbb{E}_t\left[\left\|\mathbf{z}_t - \frac{1}{B}\sum_{i\in\mathcal{B}_t}\partial_1 F_i(\mathbf{w}_t, \nu_{i,t}) + \frac{1}{B}\sum_{i\in\mathcal{B}_t}\partial_1 F_i(\mathbf{w}_t, \nu_{i,t}) - \nabla_1 F(\mathbf{w}_t, \boldsymbol{\nu}_t)\right\|_2^2\right] \\
&= \mathbb{E}_t\left[\left\|\mathbf{z}_t - \frac{1}{B}\sum_{i\in\mathcal{B}_t}\partial_1 F_i(\mathbf{w}_t, \nu_{i,t})\right\|_2^2\right] + \mathbb{E}_t\left[\left\|\frac{1}{B}\sum_{i\in\mathcal{B}_t}\partial_1 F_i(\mathbf{w}_t, \nu_{i,t}) - \nabla_1 F(\mathbf{w}_t, \boldsymbol{\nu}_t)\right\|_2^2\right] \\
&\le \frac{\sigma_1^2}{B} + \frac{\delta^2(n-B)}{B(n-1)} \coloneqq \sigma^2.
\end{aligned}$$

Due to the update of $\mathbf{w}$, we have

$$\|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2^2 = \|\mathbf{w}_t - \mathbf{w}_*\|_2^2 - 2\eta\mathbf{z}_t^\top(\mathbf{w}_t - \mathbf{w}_*) + \eta^2\|\mathbf{z}_t\|_2^2.$$

Then,

$$\begin{aligned}
&\mathbb{E}\|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2^2 \le \mathbb{E}\|\mathbf{w}_t - \mathbf{w}_*\|_2^2 - 2\eta\mathbb{E}[\nabla_1 F(\mathbf{w}_t, \boldsymbol{\nu}_t)^\top(\mathbf{w}_t - \mathbf{w}_*)] \\
&+ \eta^2\mathbb{E}\|\nabla_1 F(\mathbf{w}_t, \boldsymbol{\nu}_t)\|_2^2 + \eta^2\sigma^2.
\end{aligned} \tag{5.71}$$

Summing over $t = 0, \ldots, T-1$ and rearranging it finishes the proof. □

Lemma 5.22 *Let* $D_{\nu,0}^2 \coloneqq \mathbb{E}\|\boldsymbol{\nu}_0 - \boldsymbol{\nu}_*\|_2^2$ *and* $\gamma_t = \gamma$*, we have*

$$\frac{1}{T}\sum_{t=0}^{T-1}(2\mathbb{E}[\nabla_2 F(\mathbf{w}_t, \boldsymbol{\nu}_t)^\top(\boldsymbol{\nu}_t - \boldsymbol{\nu}_*)] - \gamma n\mathbb{E}\|\nabla_2 F(\mathbf{w}_t, \boldsymbol{\nu}_t)\|_2^2) \le \frac{D_{\nu,0}^2}{\gamma BT} + \gamma\sigma_2^2.$$

Proof. Let $\mathbb{E}_t$ denote the expectation over the random samples in the t-th iteration. Note that $\mathbb{E}_t[\nabla_2\Phi_i(\mathbf{w}_t, \nu_{i,t}; \zeta_{i,t})] = \nabla_2 F_i(\mathbf{w}_t, \nu_{i,t})$ and $\mathbb{E}_t\|\nabla_2\Phi_i(\mathbf{w}_t, \nu_{i,t}; \zeta_{i,t}) - \nabla_2 F_i(\mathbf{w}_t, \nu_{i,t})\|_2^2 \leq \sigma_0^2$ for each $i \in [n]$ (For those $i \notin \mathcal{B}_t$, $\nabla_2\Phi_i(\mathbf{w}_t, \nu_{i,t}; \zeta_{i,t})$ are not explicitly computed). For each $i \in [n]$, we have

$$\begin{aligned}
&\mathbb{E}\|\nu_{i,t+1} - \nu_{i,*}\|_2^2 \\
&= (1 - \frac{B}{n})\mathbb{E}\|\nu_{i,t} - \nu_{i,*}\|_2^2 + \frac{B}{n}\mathbb{E}\|\nu_{i,t} - \gamma\nabla_2\Phi_i(\mathbf{w}_t, \nu_{i,t}; \zeta_{i,t}) - \nu_{i,*}\|_2^2 \\
&\leq \mathbb{E}\|\nu_{i,t} - \nu_{i,*}\|_2^2 - \frac{2\gamma B}{n}\mathbb{E}[\nabla_2 F_i(\mathbf{w}_t, \nu_{i,t})^\top(\nu_{i,t} - \nu_{i,*})] + \frac{\gamma^2 B}{n}\mathbb{E}\|\nabla_2 F_i(\mathbf{w}_t, \nu_{i,t})\|_2^2 \\
&+ \frac{\gamma^2\sigma_2^2 B}{n}.
\end{aligned}$$

Summing over $i \in [n]$ leads to

$$\begin{aligned}
\mathbb{E}\|\boldsymbol{\nu}_{t+1} - \boldsymbol{\nu}_*\|_2^2 = \mathbb{E}\|\boldsymbol{\nu}_t - \boldsymbol{\nu}_*\|_2^2 - \frac{2\gamma B}{n}\mathbb{E}\Big[\sum_{i=1}^n \nabla_2 F_i(\mathbf{w}_t, \nu_{i,t})^\top(\nu_{i,t} - \nu_{i,*})\Big] \\
+ \frac{\gamma^2 B}{n}\mathbb{E}\Big[\sum_{i=1}^n \|\nabla_2 F_i(\mathbf{w}_t, \boldsymbol{\nu}_{i,t})\|_2^2\Big] + \gamma^2\sigma_2^2 B. \qquad (5.72)
\end{aligned}$$

Since

$$\begin{aligned}
\nabla_2 F(\mathbf{w}_t, \boldsymbol{\nu}_t)^\top(\boldsymbol{\nu}_t - \boldsymbol{\nu}_*) &= \frac{1}{n}\sum_{i=1}^n \nabla_2 F_i(\mathbf{w}_t, \nu_{i,t})(\nu_{i,t} - \nu_{i,*}) \\
\|\nabla_2 F(\mathbf{w}_t, \boldsymbol{\nu}_t)\|_2^2 &= \frac{1}{n^2}\sum_{i=1}^n \|\nabla_2 F_i(\mathbf{w}_t, \nu_{i,t})\|_2^2,
\end{aligned}$$

plugging these into (5.73) we have

$$\begin{aligned}
\mathbb{E}\|\boldsymbol{\nu}_{t+1} - \boldsymbol{\nu}_*\|_2^2 \leq \mathbb{E}\|\boldsymbol{\nu}_t - \boldsymbol{\nu}_*\|_2^2 - 2\gamma B\mathbb{E}\Big[\nabla_2 F(\mathbf{w}_t, \boldsymbol{\nu}_t)^\top(\boldsymbol{\nu}_t - \boldsymbol{\nu}_*)\Big] \\
+ \gamma^2 nB\mathbb{E}\Big[\|\nabla_2 F(\mathbf{w}_t, \boldsymbol{\nu}_t)\|_2^2\Big] + \gamma^2\sigma_2^2 B. \qquad (5.73)
\end{aligned}$$

Summing over $t = 0, \ldots, T-1$ and rearranging it finishes the proof. □

Theorem 5.11 (Smooth case) *Suppose Assumption 5.13, 5.14(i) and 5.15 hold. If we set* $\gamma = \min\{\frac{1}{2nL_F}, \frac{\epsilon}{2\sigma_2^2}\}$, $\eta = \min\{\frac{1}{2L_F}, \frac{\epsilon}{2\sigma^2}\}$ *and* $T = \max(\frac{2D_{\mathbf{w},0}^2}{\eta\epsilon}, \frac{2D_{\nu,0}^2}{\gamma B\epsilon})$, *then ASGD guarantees that*

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1}(F(\mathbf{w}_t, \boldsymbol{\nu}_t) - F(\mathbf{w}_*, \boldsymbol{\nu}_*))\right] \leq \epsilon.$$

The iteration complexity is

$$T = O\left(\max\left\{\frac{D_{\mathbf{w},0}^2 L_F}{\epsilon}, \frac{nD_{\nu,0}^2 L_F}{B\epsilon}, \frac{D_{\mathbf{w},0}^2 \sigma_1^2}{\epsilon^2}, \frac{D_{\nu,0}^2 \sigma_2^2}{B\epsilon^2}\right\}\right),$$

where $\sigma^2 = \frac{\sigma_1^2}{B} + \frac{\delta^2(n-B)}{B(n-1)}$.

Proof. From Lemma 5.21 and Lemma 5.22, we have

$$\frac{1}{T}\sum_{t=0}^{T-1}(2\mathbb{E}\nabla_1 F(\mathbf{w}_t, \nu_t)^\top(\mathbf{w}_t - \mathbf{w}_*) - \eta\mathbb{E}\|\nabla_1 F(\mathbf{w}_t, \nu_t)\|_2^2) \le \frac{D_{\mathbf{w},0}^2}{\eta T} + \eta\sigma^2,$$
$$\frac{1}{T}\sum_{t=0}^{T-1}(2\mathbb{E}\nabla_2 F(\mathbf{w}_t, \nu_t)^\top(\nu_t - \nu_*) - \gamma n\mathbb{E}\|\nabla_2 F(\mathbf{w}_t, \nu_t)\|_2^2) \le \frac{D_{\nu,0}^2}{\gamma BT} + \gamma\sigma_2^2.$$

If F is smooth and $\eta \le \frac{1}{2L_F}$ and $\gamma n \le \frac{1}{2L_F}$,

$$\begin{aligned}\eta\|\nabla_1 F(\mathbf{w}_t, \nu_t)\|_2^2 + \gamma n\|\nabla_2 F(\mathbf{w}_t, \nu_t)\|_2^2 &\le \frac{1}{2L_F}\left(\|\nabla_1 F(\mathbf{w}, \nu)\|_2^2 + \|\nabla_2 F(\mathbf{w}, \nu)\|_2^2\right)\\ &\le F(\mathbf{w}_t, \nu_t) - F(\mathbf{w}_*, \nu_*),\end{aligned}$$

where the last inequality uses the Lemma 1.5(b).

On the other hand, the joint convexity of F implies

$$F(\mathbf{w}_t, \nu_t) - F(\mathbf{w}_*, \nu_*) \le \nabla_1 F(\mathbf{w}_t, \nu_t)^\top(\mathbf{w}_t - \mathbf{w}_*) + \nabla_2 F(\mathbf{w}_t, \nu_t)^\top(\nu_t - \nu_*).$$

Then combining the above inequalities, we have

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1}[F(\mathbf{w}_t, \nu_t) - F(\mathbf{w}_*, \nu_*)]\right] \le \frac{D_{\mathbf{w},0}^2}{2\eta T} + \frac{\eta\sigma^2}{2} + \frac{D_{\nu,0}^2}{2\gamma BT} + \frac{\gamma\sigma_2^2}{2}.$$

In order to let the RHS above be less than ϵ, we set $\gamma = \min\{\frac{1}{2nL_F}, \frac{\epsilon}{2\sigma_2^2}\}$ and $\eta = \min\{\frac{1}{2L_F}, \frac{\epsilon}{2\sigma^2}\}$, and $T \ge \max(\frac{2D_{\mathbf{w},0}^2}{\eta\epsilon}, \frac{2D_{\nu,0}^2}{\gamma B\epsilon})$. As a result, the complexity is the in the order of

$$T = O\left(\max\left\{\frac{D_{\mathbf{w},0}^2 L_F}{\epsilon}, \frac{nD_{\nu,0}^2 L_F}{B\epsilon}, \frac{D_{\mathbf{w},0}^2 \sigma^2}{\epsilon^2}, \frac{D_{\nu,0}^2 \sigma_2^2}{B\epsilon^2}\right\}\right).$$

□

Theorem 5.12 (Non-smooth case) *Suppose Assumption 5.13, 5.14(ii) and 5.15 hold. If we set* $\gamma = \frac{\epsilon}{2(G_2^2+\sigma_2^2)}$, $\eta = \frac{\epsilon}{2(G_1^2+\sigma^2)}$ *and* $T = \max(\frac{2D_{\mathbf{w},0}^2}{\eta\epsilon}, \frac{2D_{\nu,0}^2}{\gamma B\epsilon})$, *then ASGD guarantees that*

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1}(F(\mathbf{w}_t, \nu_t) - F(\mathbf{w}_*, \nu_*))\right] \le \epsilon.$$

The iteration complexity is

$$T = O\left(\max\left\{\frac{D_{\mathbf{w},0}^2(G_1^2+\sigma^2)}{\epsilon^2}, \frac{D_{\nu,0}^2(G_2^2+\sigma_2^2)}{B\epsilon^2}\right\}\right).$$

We leave the proof as an exercise for the reader.

Why it matters

Since $F(\mathbf{w}, \nu)$ is jointly convex in $(\mathbf{w}, \nu)$, the above two theorems imply convergence of the objective with respect to the primary variable $\mathbf{w}$, i.e., $F_1(\mathbf{w}) = \min_\nu F(\mathbf{w}, \nu)$. In particular, if we define the averaged iterate $\bar{\mathbf{w}}_T = \frac{1}{T}\sum_{t=0}^{T-1}\mathbf{w}_t$, we have

$$\begin{aligned}\mathbb{E}[F_1(\bar{\mathbf{w}}_T) - F_1(\mathbf{w}_*)] &\leq \mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1}(F_1(\mathbf{w}_t) - F_1(\mathbf{w}_*))\right] \\ &\leq \mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1}(F(\mathbf{w}_t, \nu_t) - F(\mathbf{w}_*, \nu_*))\right] \leq \epsilon.\end{aligned}$$

5.5.1.2 Non-convex loss

If $s_i(\mathbf{w}, \zeta)$ is non-convex, we consider two different cases: (1) smooth case and (2) non-smooth weakly convex case. If $F(\mathbf{w}, \nu)$ is smooth in terms of $\mathbf{w}$, ν and is strongly convex in terms of ν (e.g, compositional entropic risk or COCE with χ^2 divergence for $\phi(\cdot)$), we can follow the analysis in Chapter 4 [Section 4.5] to design an algorithm and an analysis to prove the convergence for finding an ϵ-stationary point of $F_1(\mathbf{w}) = \min_\nu F(\mathbf{w}; \nu)$. We leave this as an exercise for the reader.

Below, we analyze the convergence of ASGD for non-smooth weakly convex losses. We also assume ϕ^* is non-smooth such that it covers the CCVaR minimization.

Assumption 5.16. *Suppose the following conditions hold:*

- $s_i(\mathbf{w}; \zeta)$ *is* ρ_0*-weakly convex with respect to* $\mathbf{w}$*, and* $\mathbb{E}_\zeta[\|\partial s_i(\mathbf{w}; \zeta)\|_2^2] \leq G_\ell^2$*;*
- *Assume* $|\frac{\partial \phi^*(q)}{\partial q}| \leq G_0$ *for any* $q = s_i(\mathbf{w}, \zeta) - \nu_i$*.*

Lemma 5.23 $F(\mathbf{w}, \nu)$ *is* ρ*-weakly convex with respect to* $(\mathbf{w}, \nu)$*, where* $\rho = \rho_0 G_0$*.*

Proof. We first prove that $\phi^*(s_i(\mathbf{w}; \zeta) - \nu_i)$ is weakly convex in terms of $(\mathbf{w}, \nu_i)$, i.e. there exists $\rho > 0$ such that $\phi^*(s_i(\mathbf{w}; \zeta) - \nu_i) + \frac{\rho}{2}\|\mathbf{w}\|_2^2 + \frac{\rho}{2}\nu_i^2$ is jointly convex in terms of $\mathbf{w}, \nu_i$.

Since $s_i(\mathbf{w}; \zeta)$ is ρ_0-weakly convex, we have that $q(\mathbf{w}, \nu_i, \zeta) = s_i(\mathbf{w}, \zeta) - \nu_i$ is ρ_0-weakly convex in terms of $\mathbf{v}_i = (\mathbf{w}, \nu_i)$:

$$q(\mathbf{v}_i, \zeta) \geq q(\mathbf{v}_i', \zeta) + \partial q(\mathbf{v}_i', \zeta)^\top(\mathbf{v}_i - \mathbf{v}_i') - \frac{\rho_0}{2}\|\mathbf{v}_i' - \mathbf{v}_i\|_2^2, \forall \mathbf{v}_i, \mathbf{v}_i'.$$

For any ζ, we abbreviate $q(\mathbf{v}_i;\zeta)$ as $q(\mathbf{v}_i)$. Since ϕ^* is convex and monotonically non-decreasing, for any $\omega \in \partial\phi^*(q(\mathbf{v}_i')) \in [0, G_0]$ we have

$$\begin{aligned}
&\phi^*(q(\mathbf{v}_i)) \geq \phi^*(q(\mathbf{v}_i')) + \omega(q(\mathbf{v}_i) - q(\mathbf{v}_i')) \\
&\geq \phi^*(q(\mathbf{v}_i')) + \omega(\partial q(\mathbf{v}_i')^\top(\mathbf{v}_i - \mathbf{v}_i') - \frac{\rho_0}{2}\|\mathbf{v}_i - \mathbf{v}_i'\|_2^2) \\
&\geq \phi^*(q(\mathbf{v}_i')) + \partial\phi^*(q(\mathbf{v}_i'))^\top(\mathbf{v}_i - \mathbf{v}_i') - \frac{G_0\rho_0}{2}\|\mathbf{v}_i - \mathbf{v}_i'\|_2^2.
\end{aligned}$$

The above inequality implies that $\phi^*(s_i(\mathbf{w};\zeta) - \nu_i)$ is $\rho = G_0\rho_0$-weakly convex in terms of $(\mathbf{w}, \nu_i)$, i.e., $\mathbb{E}_\zeta\phi^*(s_i(\mathbf{w};\zeta) - \nu_i) + \frac{\rho}{2}(\|\mathbf{w}\|_2^2 + |\nu_i|^2)$ is convex. As a result $F(\mathbf{w}, \boldsymbol{\nu}) + \frac{\rho}{2}\|\mathbf{w}\|_2^2 + \frac{\rho}{2}\|\boldsymbol{\nu}\|_2^2$ is jointly convex in terms of $(\mathbf{w}, \boldsymbol{\nu})$. □

Similar to the SGD for weakly convex objectives in Chapter 3[Section 3.1.4], we use the Moreau envelope of $F(\mathbf{w};\boldsymbol{\nu})$. In particular, let $\mathbf{v} = (\mathbf{w}^\top, \boldsymbol{\nu}^\top)^\top$ and consider some $\bar{\rho} > \rho$, we define:

$$F_{1/\bar{\rho}}(\mathbf{v}) = \min_{\mathbf{u}} F(\mathbf{u}) + \frac{\bar{\rho}}{2}\|\mathbf{u} - \mathbf{v}\|_2^2, \tag{5.74}$$

$$\text{prox}_{F/\bar{\rho}}(\mathbf{v}) := \arg\min_{\mathbf{u}} F(\mathbf{u}) + \frac{\bar{\rho}}{2}\|\mathbf{u} - \mathbf{v}\|_2^2. \tag{5.75}$$

Convergence Analysis

Lemma 5.24 *Under Assumption 5.16, we have*

$$\mathbb{E}_t[\|\mathbf{z}_t\|_2^2] \leq G_1^2, \quad |\partial_2 F_i(\mathbf{w}, \nu_i)|^2 \leq G_2^2,$$

where $G_1^2 = G_0^2 G_\ell^2$, *and* $G_2^2 = (1 + G_0)^2$.

Proof. For the first part,

$$\mathbb{E}_t[\|\mathbf{z}_t\|_2^2] = \mathbb{E}_t\left[\left\|\frac{1}{B}\sum_{i\in\mathcal{B}_t}\partial_1\Phi_i(\mathbf{w}_t, \nu_{i,t}; \zeta_{i,t})\right\|_2^2\right] \leq G_0^2 G_\ell^2.$$

For the second part,

$$|\partial_2 F_i(\mathbf{w}, \nu_i)|^2 = \left|\mathbb{E}_\zeta\left[-\frac{\partial\phi^*(q(\mathbf{w}, \nu_i; \zeta))}{\partial q} + 1\right]\right|^2 \leq (1 + G_0)^2.$$

□

Lemma 5.25 *Under Assumption (5.16), let* $\mathbf{v}_t = (\mathbf{w}_t^\top, \boldsymbol{\nu}_t^\top)^\top$, *for one iteration of ASGD, we have*

$$\mathbb{E}_t[F_{1/\bar{\rho}}(\mathbf{v}_{t+1})] \le F_{1/\bar{\rho}}(\mathbf{v}_t) + \bar{\rho}\eta_t(F(\bar{\mathbf{v}}_t) - F(\mathbf{v}_t) + \frac{\rho}{2}\|\mathbf{v}_t - \bar{\mathbf{v}}_t\|_2^2) + \frac{\bar{\rho}\eta_t^2(G_1^2 + G_2^2/B)}{2},$$

where $\bar{\mathbf{v}}_t = prox_{F/\bar{\rho}}(\mathbf{v}_t)$.

Proof. Let $\mathbb{E}_t$ denote the expectation over the random samples at the t-th iteration conditioned on that in all previous iterations.

$$\begin{aligned}
&\mathbb{E}_t[F_{1/\bar{\rho}}(\mathbf{v}_{t+1})] \le \mathbb{E}_t\left[F(\bar{\mathbf{v}}_t) + \frac{\bar{\rho}}{2}\|\mathbf{v}_{t+1} - \bar{\mathbf{v}}_t\|_2^2\right] \\
&\le F(\bar{\mathbf{v}}_t) + \frac{\bar{\rho}}{2}\mathbb{E}_t[\|\mathbf{w}_t - \eta_t\mathbf{z}_t - \bar{\mathbf{w}}_t\|_2^2 + \|\boldsymbol{\nu}_{t+1} - \bar{\boldsymbol{\nu}}_t\|_2^2] \\
&\le F(\bar{\mathbf{v}}_t) + \frac{\bar{\rho}}{2}\mathbb{E}_t[\|\mathbf{w}_t - \eta_t\mathbf{z}_t - \bar{\mathbf{w}}_t\|_2^2] + \frac{\bar{\rho}}{2}\mathbb{E}_t[\|\boldsymbol{\nu}_{t+1} - \bar{\boldsymbol{\nu}}_t\|_2^2] \\
&\le F(\bar{\mathbf{v}}_t) + \frac{\bar{\rho}}{2}\|\mathbf{w}_t - \bar{\mathbf{w}}_t\|_2^2 + \bar{\rho}\eta_t\mathbb{E}_t[(\bar{\mathbf{w}}_t - \mathbf{w}_t)^\top\partial_1 F(\mathbf{w}_t, \boldsymbol{\nu}_t)] + \frac{\bar{\rho}\eta_t^2 G_1^2}{2} \\
&+ \frac{\bar{\rho}}{2}\mathbb{E}_t[\|\boldsymbol{\nu}_{t+1} - \bar{\boldsymbol{\nu}}_t\|_2^2]
\end{aligned}$$

where the last step uses $\mathbb{E}_t[\mathbf{z}_t] = \partial_1 F(\mathbf{w}_t, \boldsymbol{\nu}_t)$ and $\mathbb{E}[\|\mathbf{z}_t\|_2^2] \le G_1^2$.

Similar to (5.73), we can prove that

$$\mathbb{E}_t\|\boldsymbol{\nu}_{t+1} - \bar{\boldsymbol{\nu}}_t\|_2^2 = \|\boldsymbol{\nu}_t - \bar{\boldsymbol{\nu}}_t\|_2^2 - 2\gamma_t B\partial_2 F(\mathbf{w}_t, \boldsymbol{\nu}_t)^\top(\boldsymbol{\nu}_t - \bar{\boldsymbol{\nu}}_t) + \gamma_t^2 G_2^2 B.$$

Let $\gamma_t B = \eta_t$, combining the above we have

$$\begin{aligned}
&\mathbb{E}_t[F_{1/\bar{\rho}}(\mathbf{v}_{t+1})] \\
&\le F(\bar{\mathbf{v}}_t) + \frac{\bar{\rho}}{2}\|\mathbf{v}_t - \bar{\mathbf{v}}_t\|_2^2 + \bar{\rho}\eta_t\mathbb{E}_t[(\bar{\mathbf{v}}_t - \mathbf{v}_t)^\top\partial F(\mathbf{v}_t)] + \frac{\bar{\rho}\eta_t^2(G_1^2 + G_2^2/B)}{2} \\
&\le F(\bar{\mathbf{v}}_t) + \frac{\bar{\rho}}{2}\|\mathbf{v}_t - \bar{\mathbf{v}}_t\|_2^2 + \bar{\rho}\eta_t\mathbb{E}_t[(\bar{\mathbf{v}}_t - \mathbf{v}_t)^\top\partial F(\mathbf{v}_t)] + \frac{\bar{\rho}\eta_t^2(G_1^2 + G_2^2/B)}{2} \\
&\le F_{1/\bar{\rho}}(\mathbf{v}_t) + \bar{\rho}\eta_t(F(\bar{\mathbf{v}}_t) - F(\mathbf{v}_t) + \frac{\rho}{2}\|\mathbf{v}_t - \bar{\mathbf{v}}_t\|_2^2) + \frac{\bar{\rho}\eta_t^2(G_1^2 + G_2^2/B)}{2}.
\end{aligned}$$

where the last step uses the definition of $F_{1/\bar{\rho}}(\mathbf{v}_t)$ and the ρ-weak convexity of F. Rearranging this inequality finishes the proof.

□

Theorem 5.13 *Suppose Assumption (5.16) holds and* $F_* = \inf F(\mathbf{w}, \boldsymbol{\nu}) \ge \infty$, *by setting* $\bar{\rho} = 2\rho$, $\eta = \epsilon^2/(2\bar{\rho}(G_1^2 + G_2^2/B))$, $\gamma = \eta/B$ *and* $T \ge \frac{4(F(\mathbf{w}_0, \boldsymbol{\nu}_0) - F_*)}{\epsilon^2\eta}$, *ASGD guarantees that*

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\|\nabla F_{1/\bar{\rho}}(\mathbf{v}_t)\|_2^2\right] \le \epsilon^2$$

with a complexity of $T = O\left(\frac{\rho(G_1^2+G_2^2/B)}{\epsilon^4}\right)$.

Proof. Since $F(\mathbf{v}) + \frac{\bar{\rho}}{2}\|\mathbf{v} - \mathbf{v}_t\|_2^2$ is $(\bar{\rho} - \rho)$-strongly convex and have a minimum solution at $\bar{\mathbf{v}}_t$, then we have

$$\begin{aligned}
&F(\mathbf{v}_t) - F(\bar{\mathbf{v}}_t) - \frac{\rho}{2}\|\mathbf{v}_t - \bar{\mathbf{v}}_t\|_2^2 \\
&= (F(\mathbf{v}_t) + \frac{\bar{\rho}}{2}\|\mathbf{v}_t - \mathbf{v}_t\|_2^2) - (F(\bar{\mathbf{v}}_t) + \frac{\bar{\rho}}{2}\|\bar{\mathbf{v}}_t - \mathbf{v}_t\|_2^2) + (\frac{\bar{\rho}}{2} - \frac{\rho}{2})\|\mathbf{v}_t - \bar{\mathbf{v}}_t\|_2^2 \\
&\geq \frac{(\bar{\rho} - \rho)}{2}\|\mathbf{v}_t - \bar{\mathbf{v}}_t\|_2^2 + \frac{(\bar{\rho} - \rho)}{2}\|\mathbf{v}_t - \bar{\mathbf{v}}_t\|_2^2 = (\bar{\rho} - \rho)\|\mathbf{v}_t - \bar{\mathbf{v}}_t\|_2^2 \\
&= \frac{\bar{\rho} - \rho}{\bar{\rho}^2}\|\nabla F_{1/\bar{\rho}}(\mathbf{v}_t)\|_2^2.
\end{aligned}$$

Combining this result with that in Lemma 5.25 and noting that $\bar{\rho} = 2\rho, \eta_t = \eta$, we have

$$\begin{aligned}
\mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1}\|\nabla F_{1/\bar{\rho}}(\mathbf{v}_t)\|_2^2\right] &\leq \frac{2(F_{1/\bar{\rho}}(\mathbf{v}_0) - F_*)}{\eta T} + \bar{\rho}\eta(G_1^2 + G_2^2/B) \\
&\leq \frac{2(F(\mathbf{v}_0) - F_*)}{\eta T} + \bar{\rho}\eta(G_1^2 + G_2^2/B)
\end{aligned}$$

By setting $\eta = \epsilon^2/(2\bar{\rho}(G_1^2+G_2^2/B))$ and $T \geq \frac{4(F(\mathbf{v}_0)-F_*)}{\epsilon^2\eta}$, we have $\mathbb{E}[\|\nabla F_{1/\bar{\rho}}(\mathbf{v}_\tau)\|_2^2] \leq \epsilon^2$ for a randomly selected $\tau \in \{0, \ldots, T-1\}$. □

5.5.2 A Geometry-aware Algorithm for Entropic Risk

Although last section presents a general algorithm for solving COCE risk minimization, it may exhibits numerical instability issue and slow convergence when solving compositional entropic risk minimization:

$$\begin{aligned}
&\min_{\mathbf{w}}\min_{\nu}\left[F(\mathbf{w}, \nu) = \frac{1}{n}\sum_{i=1}^{n}\{\mathbb{E}_\zeta \exp(s_i(\mathbf{w}; \zeta) - \nu_i) - 1 + \nu_i\}\right] \\
&= \min_{\mathbf{w}}\frac{1}{n}\sum_{i=1}^{n}\log\left(\mathbb{E}_\zeta \exp(s_i(\mathbf{w}; \zeta))\right).
\end{aligned}$$

The numerical instability issue is caused by dealing with exponential functions, e.g., $\exp(s_i(\mathbf{w}; \zeta) - \nu_i)$, in calculation of stochastic gradients of ν_i. The slow convergence arises because the standard SGD update for ν_i fails to exploit the geometric structure of the problem.

5.5.2.1 Stochastic Optimization of Log-E-Exp

We first consider a simplified problem where there is only one component $n = 1$, i.e.,

$$\min_{\mathbf{w}} F_1(\mathbf{w}) := \log\left(\mathbb{E}_\zeta \exp(s(\mathbf{w};\zeta))\right). \tag{5.76}$$

The KL-regularized DRO problem (2.14) is a special case. It is also known as log-E-Exp, a more general form of the log-Sum-Exp function, where the middle "E" denotes an expectation and highlights the associated computational challenges.

Application of SCGD

At the beginning of Section 4.1, we treat this problem as a special case of stochastic compositional optimization (SCO), where the outer function is $f(\cdot) = \log(\cdot)$ and the inner function is $g(\mathbf{w}) = \mathbb{E}_\zeta[\exp(s(\mathbf{w};\zeta))]$. Let us first apply the SCGD algorithm. The key updates are presented below:

$$\begin{aligned} u_t &= (1-\gamma_t)u_{t-1} + \gamma_t \exp(s(\mathbf{w}_t;\zeta_t)), \\ \mathbf{z}_t &= \frac{1}{u_t}\exp(s(\mathbf{w}_t;\zeta_t'))\nabla s(\mathbf{w}_t;\zeta_t'), \\ \mathbf{w}_{t+1} &= \mathbf{w}_t - \eta_t \mathbf{z}_t, \end{aligned} \tag{5.77}$$

where u_t is an estimator of the inner function value $g(\mathbf{w}_t)$ and $\mathbf{z}_t = \nabla f(u_t)\nabla g(\mathbf{w}_t;\zeta_t')$ is a gradient estimator of $\mathbf{w}_t$.

From a practitioner's perspective, the algorithm can be readily implemented and applied to real applications. However, from a theoretical perspective, several open problems remain. In particular: (1) Can we establish an $O(1/\epsilon^2)$ convergence rate for this algorithm to find an ϵ-optimal solution when $s(\mathbf{w};\zeta)$ is convex? (2) If yes, what are the practical advantages of this algorithm compared with the ASGD method presented in the previous section?

Wait! Shouldn't we established the convergence rate of SCGD in Chapter 4? It is true that we presented a convergence analysis of the above algorithm for non-convex problems under proper conditions, however, it remains an open problem to establish the complexity of $O(1/\epsilon^2)$ for finding an ϵ-optimal solution under the convexity of $s(\mathbf{w};\zeta)$. A naive analysis of SCGD for convex problems yields a complexity of $O(1/\epsilon^4)$ (see Wang et al. (2017a)).

A Novel Algorithm

To address these open questions, we present a novel algorithm based on the min-min reformulation of log-E-exp, i.e.,

$$\min_{\mathbf{w}} \min_{\nu} F(\mathbf{w}, \nu) := \mathbb{E}_{\zeta} \exp(s(\mathbf{w}; \zeta) - \nu) + \nu. \tag{5.78}$$

where we ignored the constant -1 in the objective. As proved in Lemma 5.20, $F(\mathbf{w}; \nu)$ is jointly convex in terms of $\mathbf{w}, \nu$ when $s(\mathbf{w}; \zeta)$ is convex.

Motivation

The key novelty of our design is a **geometry-aware algorithm** for solving the equivalent min-min optimization (5.78). Let us first discuss the motivation. One challenge for solving the min-min optimization problem is that the objective function $F(\mathbf{w}, \nu)$ could have exponentially large smoothness constant in terms of ν. We will formally analyze this phenomenon in next section. Hence, a vanilla gradient method that uses the firs-order approximation of F will inevitably impacted by the large smoothness parameter.

To mitigate the adverse effects of a large smoothness parameter with respect to ν, we resort to the classical approach of employing a proximal mapping. Proximal mappings have been widely used to handle a non-smooth function in composite objectives consisting of a smooth loss and a non-smooth regularizer. This approach enables optimization algorithms to retain the favorable convergence properties of smooth optimization and often leads to faster convergence despite the presence of non-smooth terms. Analogously, even when a function is smooth but characterized by a very large smoothness parameter, applying its proximal mapping can effectively alleviate the negative impact of this large smoothness constant.

However, there is an important distinction from classical proximal methods, which typically rely on direct access to the function of interest for computing the proximal mapping. In our setting, we cannot directly apply the proximal mapping of $F(\mathbf{w}, \nu)$. Instead, we only have access to a stochastic estimator

$$\Phi(\mathbf{w}, \nu; \zeta) = e^{s(\mathbf{w};\zeta)-\nu} + \nu,$$

defined for a random sample ζ. As a result, it becomes necessary to explicitly account for the noise introduced by this stochastic approximation.

Algorithm

To account for the stochastic noise, we introduce a Bregman divergence $D_{\varphi}(\cdot, \cdot)$ and update ν_t according to the following scheme:

$$\nu_t = \arg\min_{\nu} \Phi(\mathbf{w}_t, \nu; \zeta_t) + \frac{1}{\alpha_t} D_{\varphi}(\nu, \nu_{t-1}), \tag{5.79}$$

where $\zeta_t \sim \mathbb{P}$ is a random sample and $\alpha_t > 0$ is a step size parameter. We refer to this step as **stochastic proximal mirror descent (SPMD)** update. To respect the geometry of the stochastic objective $\Phi(\mathbf{w}_t, \nu; \zeta_t)$, we construct a tailored Bregman divergence induced by the function $\varphi(\nu) = e^{-\nu}$, namely,

Algorithm 21 The SCENT Algorithm for solving Log-E-Exp (5.76)

1: Initialize $\mathbf{w}_1$, ν_0, step sizes η_t and α_t, $\varphi(\nu) = e^{-\nu}$.
2: **for** $t = 1 \ldots, T-1$ **do**
3: Sample ζ_t, ζ_t'
4: Update $\nu_t = \arg\min_\nu \exp(s(\mathbf{w}_t; \zeta_t) - \nu) + \nu + \frac{1}{\alpha_t} D_\varphi(\nu, \nu_{t-1})$
5: Compute $\mathbf{z}_t = \exp(s(\mathbf{w}_t; \zeta_t') - \nu_t) \nabla s(\mathbf{w}_t; \zeta_t')$
6: Compute $\mathbf{v}_t = (1 - \beta_t)\mathbf{v}_{t-1} + \beta_t \mathbf{z}_t$
7: Update $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{v}_t$
8: **end for**

$$D_\varphi(\nu, \nu_{t-1}) = e^{-\nu} - e^{-\nu_{t-1}} + e^{-\nu_{t-1}}(\nu - \nu_{t-1}). \tag{5.80}$$

Once we have ν_t, we compute a vanilla gradient estimator by

$$\mathbf{z}_t = \exp(s(\mathbf{w}_t; \zeta_t') - \nu_t) \nabla s(\mathbf{w}_t; \zeta_t'). \tag{5.81}$$

If the problem is non-convex, we compute a moving-average estimator $\mathbf{v}_t = (1 - \beta_t)\mathbf{v}_{t-1} + \beta_t \mathbf{z}_t$ and then update the model parameter $\mathbf{w}_{t+1}$. We present the full steps in Algorithm 21, which is referred to SCENT.

SCGD is just a special case of SCENT

To see the connection with SCGD, we present the following lemma.

Lemma 5.26 *The update of ν_t defined by (5.79) can be computed by*

$$e^{\nu_t} = \frac{1}{1 + \alpha_t e^{\nu_{t-1}}} e^{\nu_{t-1}} + \frac{\alpha_t e^{\nu_{t-1}}}{1 + \alpha_t e^{\nu_{t-1}}} \exp(s(\mathbf{w}_t; \zeta_t)). \tag{5.82}$$

If $y_t = e^{-\nu_t}$, we have

$$y_t = \frac{y_{t-1} + \alpha_t}{1 + \alpha_t e^{s(\mathbf{w}_t; \zeta_t)}}.$$

Proof. We compute the gradient of the problem (5.79) and set it to zero for computing ν_t, i.e.,

$$-\exp(s(\mathbf{w}_t; \zeta_t) - \nu_t) + 1 + \frac{1}{\alpha_t}(-e^{-\nu_t} + e^{-\nu_{t-1}}) = 0.$$

Solving this equation finishes the proof. □

If we define $u_t = e^{\nu_t}$ and $\gamma_t' = \frac{\alpha_t e^{\nu_{t-1}}}{1 + \alpha_t e^{\nu_{t-1}}}$, then the updates of SCENT ($\beta_t = 1$) are equivalent to

$$
\begin{aligned}
u_t &= (1-\gamma_t')u_{t-1} + \gamma_t' \exp(s(\mathbf{w}_t;\zeta_t)) \\
\mathbf{z}_t &= \frac{1}{u_t}\exp(s(\mathbf{w}_t;\zeta_t'))\nabla s(\mathbf{w}_t;\zeta_t'), \\
\mathbf{w}_{t+1} &= \mathbf{w}_t - \eta_t \mathbf{z}_t.
\end{aligned} \tag{5.83}
$$

Comparing this update with that of SCGD (5.77), the key difference lies in the choice of the moving-average parameter: SCENT adopts an adaptive parameter $\gamma_t' = \frac{\alpha_t e^{\nu_{t-1}}}{1+\alpha_t e^{\nu_{t-1}}}$, whereas SCGD uses a non-adaptive γ_t. If we set $\alpha_t = \frac{\gamma_t}{1-\gamma_t} e^{-\nu_{t-1}}$, then the updates of SCENT reduce to that of SCGD.

Convergence analysis for convex problems

Since $\mathbf{z}_t = \nabla_{\mathbf{w}} \exp(s(\mathbf{w}_t;\zeta_t') - \nu_t)$, we have

$$
\mathbb{E}_{\zeta_t'}[\mathbf{z}_t] = \nabla_{\mathbf{w}} \mathbb{E}_{\zeta_t'}[\exp(s(\mathbf{w}_t;\zeta_t') - \nu_t)] = \nabla_1 F(\mathbf{w}_t, \nu_t).
$$

Let $\mathbf{w}_*, \nu_*$ be the optimal solution:

$$
(\mathbf{w}_*, \nu_*) = \arg\min_{\mathbf{w},\nu} F(\mathbf{w}, \nu).
$$

It is straightforward to derive $\nu_* = \log[\mathbb{E}\exp(s(\mathbf{w}_*;\zeta))]$.

Assumption 5.17. *Assume that the following conditions hold:*

(i) $s(\mathbf{w};\zeta)$ is convex;
(ii) the loss function is bounded such that $s(\mathbf{w};\zeta) \in [c_0, c_1], \forall \mathbf{w}, \zeta$.
(iii) there exists G such that $\mathbb{E}_\zeta \|\nabla s(\mathbf{w}_t, \zeta)\|_2^2] \le G^2, \forall t$.

Critical: To relax the second assumption, we can assume that $\mathbf{w}$ is restricted to a bounded domain $\mathcal{W}$ and $s(\mathbf{w};\zeta)$ is regular. In practice, we always enforce the boundness of $\mathbf{w}_t$ through either projection onto $\mathcal{W}$ or using a regularizer $r(\mathbf{w})$. The update of $\mathbf{w}_{t+1}$ can be modified as the SPGD update:

$$
\mathbf{w}_{t+1} = \arg\min_{\mathbf{w}} \mathbf{z}_t^\top \mathbf{w} + r(\mathbf{w}) + \frac{1}{\eta_t}\|\mathbf{w} - \mathbf{w}_t\|_2^2.
$$

The analysis can be performed similarly.

Lemma 5.27 *Under Assumption 5.17(ii), $\nu_* \in [c_0, c_1]$ and if $\nu_0 \in [c_0, c_1]$ then $\nu_t \in [c_0, c_1], \forall t$.*

Proof. $\nu_* \in [c_0, c_1]$ can be seen from $\nu_* = \log[\mathbb{E}\exp(s(\mathbf{w}_*;\zeta))]$. The second result can be easily seen from the update of e^{ν_t} as in (5.82) by induction. □

For the ease of analysis, we define two quantities to capture the variance terms caused by using stochastic estimators.

$$\sigma_t^2 := \mathbb{E}_{\zeta'}\|\exp(s(\mathbf{w}_t;\zeta_t') - \nu_t)\nabla s(\mathbf{w}_t;\zeta_t')\|_2^2],$$
$$\delta_t^2 := \mathbb{E}_{\zeta_t}[e^{-\nu_{t-1}}|e^{s(\mathbf{w}_t;\zeta_t)} - \mathbb{E}_{\zeta}[e^{s(\mathbf{w}_t;\zeta)}]|^2].$$

Under Assumption 5.17 (ii) and (iii), σ_t, δ_t are bounded because $e^{\nu_t}, e^{\nu_{t-1}}$ and $e^{s(\mathbf{w}_t;\zeta_t)}$ is upper and lower bounded.

Critical: These two quantities are related to the variance of stochastic estimators in terms of $\mathbf{w}_t$ and ν_t, respectively. Both quantities have a normalization term $e^{-\nu_t}$ or $e^{-\nu_{t-1}}$.

Lemma 5.28 *Under Assumption 5.17 and $\beta_t = 1$, we have*

$$\mathbb{E}[\eta_t \nabla_1 F(\mathbf{w}_t, \nu_t)^\top (\mathbf{w}_t - \mathbf{w}_*)] \leq \mathbb{E}\left[\frac{1}{2}\|\mathbf{w}_t - \mathbf{w}_*\|_2^2 - \frac{1}{2}\|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2^2\right] + \frac{\eta_t^2\sigma_t^2}{2}.$$

Proof. The proof is a simple application of Lemma 3.3. □

If the SPGD update is used, we can use Lemma 3.6 giving us

$$\mathbf{z}_t^\top(\mathbf{w}_{t+1} - \mathbf{w}_*) + r(\mathbf{w}_{t+1}) - r(\mathbf{w}_*) \leq \frac{1}{2\eta_t}(\|\mathbf{w}_t - \mathbf{w}_*\|_2^2 - \|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2^2)$$
$$- \frac{1}{2\eta_t}\|\mathbf{w}_t - \mathbf{w}_{t+1}\|_2^2.$$

Then,

$$\mathbf{z}_t^\top(\mathbf{w}_t - \mathbf{w}_*) + r(\mathbf{w}_t) - r(\mathbf{w}_*) \leq \frac{1}{2\eta_t}(\|\mathbf{w}_t - \mathbf{w}_*\|_2^2 - \|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2^2)$$
$$+ \mathbf{z}_t^\top(\mathbf{w}_t - \mathbf{w}_{t+1}) - \frac{1}{2\eta_t}\|\mathbf{w}_t - \mathbf{w}_{t+1}\|_2^2 + r(\mathbf{w}_t) - r(\mathbf{w}_{t+1})$$
$$\leq \frac{1}{2\eta_t}(\|\mathbf{w}_t - \mathbf{w}_*\|_2^2 - \|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2^2) + \frac{\eta_t}{2}\|\mathbf{z}_t\|_2^2 + r(\mathbf{w}_t) - r(\mathbf{w}_{t+1}).$$

Taking expectation on both sides, we have

$$\mathbb{E}[\eta_t \nabla_1 F(\mathbf{w}_t, \nu_t)^\top (\mathbf{w}_t - \mathbf{w}_*)] + \eta_t(r(\mathbf{w}_t) - r(\mathbf{w}_*))$$
$$\leq \mathbb{E}\left[\left(\eta_t r(\mathbf{w}_t) + \frac{1}{2}\|\mathbf{w}_t - \mathbf{w}_*\|_2^2\right) - \left(\eta_t r(\mathbf{w}_{t+1}) + \frac{1}{2}\|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2^2\right)\right] + \frac{\eta_t^2\sigma_t^2}{2}.$$

If $\eta_{t+1} \leq \eta_t$ and $r(\mathbf{w}) \geq 0$, then $\eta_t r(\mathbf{w}_{t+1}) \leq \eta_{t+1} r(\mathbf{w}_{t+1})$, then the terms in the square bracket will form a telescoping series over $t = 1, \ldots, T$. As a result, the following analysis will proceed similarly.

Lemma 5.29 *Under Assumption 5.17 (ii), we have*

$$\alpha_t \nabla_2 \Phi(\mathbf{w}_t, \nu_t; \zeta_t)^\top (\nu_t - \nu_*) \le D_\varphi(\nu_*, \nu_{t-1}) - D_\varphi(\nu_*, \nu_t) - D_\varphi(\nu_t, \nu_{t-1}).$$

Proof. Recall the definition

$$\begin{aligned}
&\Phi(\mathbf{w}_t, \nu; \zeta_t) = \exp(s(\mathbf{w}_t; \zeta_t) - \nu) + \nu \\
&\varphi(\nu) = e^{-\nu}, \quad D_\varphi(a, b) = \varphi(a) - \varphi(b) - \langle \nabla\varphi(b), a - b\rangle,
\end{aligned}$$

and the update of ν_t:

$$\nu_t = \arg\min_{\nu} \alpha_t \Phi(\mathbf{w}_t, \nu; \zeta_t) + D_\varphi(\nu, \nu_{t-1}).$$

The first-order optimality gives

$$\alpha_t \nabla_2 \Phi(\mathbf{w}_t, \nu_t; \zeta_t) + \nabla\varphi(\nu_t) - \nabla\varphi(\nu_{t-1}) = 0.$$

Taking inner product with $(\nu_t - \nu_*)$ and rearranging gives

$$\begin{aligned}
\alpha_t \langle \nabla_2 \Phi(\mathbf{w}_t, \nu_t; \zeta_t), \nu_t - \nu_* \rangle &= \langle \nabla\varphi(\nu_{t-1}) - \nabla\varphi(\nu_t), \nu_t - \nu_* \rangle \\
&= D_\varphi(\nu_*, \nu_{t-1}) - D_\varphi(\nu_*, \nu_t) - D_\varphi(\nu_t, \nu_{t-1})
\end{aligned}$$

where the last equality holds by three-point identity as in Lemma 3.9. □

Critical: To proceed the analysis, we need to bound $\mathbb{E}[\alpha_t \nabla_2 F(\mathbf{w}_t, \nu_t)^\top (\nu_t - \nu_*)]$. In light of the above lemma, we will bound the following difference in expectation:

$$\mathbb{E}[(\nabla_2 F(\mathbf{w}_t, \nu_t)^\top (\nu_t - \nu_*) - \nabla_2 \Phi(\mathbf{w}_t, \nu_t; \zeta_t)^\top (\nu_t - \nu_*)].$$

The challenge lies at ν_t depends on ζ_t, making the above expectation not equal to zero.

Lemma 5.30 *Assume $\alpha_t \le \rho e^{-\nu_{t-1}}$ for any constant $\rho > 0$, then we have*

$$|\mathbb{E}[(\nabla_2 F(\mathbf{w}_t, \nu_t)^\top (\nu_t - \nu_*) - \nabla_2 \Phi(\mathbf{w}_t, \nu_t; \zeta_t)^\top (\nu_t - \nu_*)]| \le \alpha_t \delta_t^2 C. \tag{5.84}$$

where $C = (1 + \rho)(1 + c_1 - c_0)$.

Proof. In the following proof, we let $\mathcal{F}_{t-1}$ denote the filtration (the "information available") up to time $t - 1$.

Let us define $z_t = e^{s(\mathbf{w}_t; \zeta_t)}$, $m_t = \mathbb{E}_\zeta[e^{s(\mathbf{w}_t; \zeta)} | \mathcal{F}_{t-1}]$, and $y_t = e^{-\nu_t}$. Let z and z' two independent variables so that $\mathbb{E}[z|\mathcal{F}_{t-1}] = \mathbb{E}[z'|\mathcal{F}_{t-1}] = m_t$. Since ν_t depends on z_t, let us define random functions:

$$y_t(z) = \frac{y_{t-1} + \alpha_t}{\alpha_t z + 1}, \quad \nu_t(z) = -\log y_t(z)$$
$$h_t(z) = e^{-\nu_t(z)}\big(\nu_t(z) - \nu_*\big) = y_t(z)\big(\nu_t(z) - \nu_*\big).$$

According to the update of ν_t, we have $y_t = y_t(z_t), \nu_t = \nu_t(z)$. For the target, we have

$$\begin{aligned}
&\mathbb{E}[(\nabla_2\Phi(\mathbf{w}_t, \nu_t; \zeta_t) - \nabla_2 F(\mathbf{w}_t, \nu_t))^\top (\nu_t - \nu_*) \mid \mathcal{F}_{t-1}] \\
&= \mathbb{E}[\mathbb{E}_\zeta[e^{s(\mathbf{w}_t;\zeta)}] - e^{s(\mathbf{w}_t;\zeta_t)})e^{-\nu_t}\big(\nu_t - \nu_*\big) \mid \mathcal{F}_{t-1}] \\
&= \mathbb{E}[(m_t - z_t)h_t(z_t) \mid \mathcal{F}_{t-1}] = \mathbb{E}_z[(m_t - z)h_t(z)|\mathcal{F}_{t-1}].
\end{aligned} \tag{5.85}$$

Since z' is an i.i.d. copy of z and independent of z given $\mathcal{F}_{t-1}$,

$$m_t = \mathbb{E}[z \mid \mathcal{F}_{t-1}] = \mathbb{E}[z' \mid \mathcal{F}_{t-1}].$$

Using the conditional independence,

$$\mathbb{E}\big[(m_t - z)h_t(z) \mid \mathcal{F}_{t-1}\big] = \mathbb{E}\big[(z' - z)h_t(z) \mid \mathcal{F}_{t-1}\big].$$

By exchangeability of (z, z') conditional on $\mathcal{F}_{t-1}$,

$$\mathbb{E}\big[(z' - z)h_t(z') \mid \mathcal{F}_{t-1}\big] = -\mathbb{E}\big[(z' - z)h_t(z) \mid \mathcal{F}_{t-1}\big].$$

Averaging the last two displays gives the standard symmetrization:

$$\mathbb{E}\big[(m_t - z)h_t(z) \mid \mathcal{F}_{t-1}\big] = \frac{1}{2}\mathbb{E}\big[(z' - z)\big(h_t(z) - h_t(z')\big) \mid \mathcal{F}_{t-1}\big]. \tag{5.86}$$

Next, we show that $h(z)$ is Lipschitz continuous. By definition,

$$y_t(z) = \frac{y_{t-1} + \alpha_t}{\alpha_t z + 1}, \quad h_t(z) = y_t(z)\big(\nu_t(z) - \nu_*\big).$$

Differentiate with respect to z:

$$\frac{dy_t(z)}{dz} = (y_{t-1} + \alpha_t)\,\frac{d}{dz}\big((\alpha_t z + 1)^{-1}\big) = -\frac{\alpha_t(y_{t-1} + \alpha_t)}{(\alpha_t z + 1)^2}.$$

Using $y_t(z)(\alpha_t z + 1) = y_{t-1} + \alpha_t$, we can rewrite this as

$$\frac{dy_t(z)}{dz} = -\frac{\alpha_t y_t(z)}{\alpha_t z + 1}.$$

Since $\nu_t(z) = -\log y_t(z)$, we have

$$\frac{d\nu_t(z)}{dz} = -\frac{1}{y_t(z)}\frac{dy_t(z)}{dz} = \frac{\alpha_t}{\alpha_t z + 1}.$$

As a result,

$$\frac{dh_t(z)}{dz} = \frac{dy_t(z)}{dz}\big(\nu_t(z) - \nu_*\big) + y_t(z)\frac{d\nu_t(z)}{dz} = \frac{\alpha_t y_t(z)}{\alpha_t z + 1}\big(1 - (\nu_t(z) - \nu_*)\big).$$

Since $\nu_t(z), \nu_* \in [c_0, c_1]$, then

$$\big|1 - (\nu_t(z) - \nu_*)\big| \le 1 + c_1 - c_0,$$

and since $y_t(z) = \frac{y_{t-1}+\alpha_t}{\alpha_t z+1} \le y_{t-1} + \alpha_t \le (1+\rho)y_{t-1}$, we have

$$\left|\frac{dh_t}{dz}\right| \le \alpha_t y_{t-1}(1+\rho)(1 + c_1 - c_0),$$

which means i.e. h_t is L_t-Lipschitz with

$$L_t \le \alpha_t y_{t-1} C.$$

Then, it holds

$$\big|(z' - z)\big(h_t(z) - h_t(z')\big)\big| \le L_t\,(z' - z)^2 \le C\alpha_t y_{t-1}(z' - z)^2.$$

Thus,

$$\begin{aligned}&\mathbb{E}\Big[\big|(z' - z)\big(h_t(z) - h_t(z')\big) \mid \mathcal{F}_{t-1}\Big] \le C\alpha_t \mathbb{E}[y_{t-1}(z' - z)^2) \mid \mathcal{F}_{t-1}] \\ &= C\alpha_t \cdot 2\mathbb{E}[y_{t-1}(z - \mathbb{E}[z])^2 \mid \mathcal{F}_{t-1}] \le 2C\alpha_t \delta_t^2,\end{aligned}$$

where the last step uses the definition of δ_t^2. Applying this result to (5.86), we have

$$\Big|\mathbb{E}\big[(\mu_t - z)h_t(z) \mid \mathcal{F}_{t-1}\big]\Big| \le \frac{1}{2}\mathbb{E}\Big[\big|(z' - z)\big(h_t(z) - h_t(z')\big)\big| \mid \mathcal{F}_{t-1}\Big] \le C\alpha_t \delta_t^2.$$

By noting (5.85), we finish the proof. □

Combining Lemma 5.29 and Lemma 5.30, we have the following lemma for one-step analysis of the ν-update.

Lemma 5.31 *Under Assumption (5.17) (ii), we have*

$$\mathbb{E}[\alpha_t \nabla_2 F(\mathbf{w}_t, \nu_t)^\top (\nu_t - \nu_*)] \le \mathbb{E}[D_\varphi(\nu_*, \nu_{t-1}) - D_\varphi(\nu_*, \nu_t) + C\alpha_t^2 \delta_t^2]. \quad (5.87)$$

Finally, we state the convergence result of SCENT in the following theorem.

Theorem 5.14 *Suppose Assumption 5.17 holds. Let $\beta_t = 1, \eta_t = \eta\alpha_t$, $\alpha_t < \rho e^{-v_{t-1}}$, then SCENT guarantees that*

$$\mathbb{E}\left[\sum_{t=1}^{T}\alpha_t(F(\mathbf{w}_t,\nu_t)-F(\mathbf{w}_*,\nu_*))\right]$$
$$\leq \frac{1}{2\eta}\|\mathbf{w}_1-\mathbf{w}\|_2^2+D_\varphi(\nu_*,\nu_0)+\mathbb{E}\left[\sum_{t=1}^{T}\frac{\eta\alpha_t^2\sigma_t^2}{2}+\sum_{t=1}^{T}C\alpha_t^2\delta_t^2\right].$$

Proof. Since $\eta_t=\eta\alpha_t$, from Lemma 5.28, we obtain

$$\mathbb{E}[\alpha_t\nabla_1 F(\mathbf{w}_t,\nu_t)^\top(\mathbf{w}_t-\mathbf{w}_*)]\leq\mathbb{E}\left[\frac{1}{2\eta}\|\mathbf{w}_t-\mathbf{w}\|_2^2-\frac{1}{2\eta}\|\mathbf{w}_{t+1}-\mathbf{w}_*\|_2^2+\frac{\eta\alpha_t^2\sigma_t^2}{2}\right].$$

Combining this with Lemma 5.31, we have

$$\mathbb{E}[\alpha_t(\nabla_1 F(\mathbf{w}_t,\nu_t)^\top(\mathbf{w}_t-\mathbf{w}_*)+\nabla_2 F(\mathbf{w}_t,\nu_t)^\top(\nu_t-\nu_*))]$$
$$\leq\mathbb{E}\left[\frac{1}{2\eta}\|\mathbf{w}_t-\mathbf{w}\|_2^2-\frac{1}{2\eta}\|\mathbf{w}_{t+1}-\mathbf{w}_*\|_2^2+D_\varphi(\nu_*,\nu_{t-1})-D_\varphi(\nu_*,\nu_t)\right]$$
$$+\mathbb{E}\left[\frac{\eta\alpha_t^2\sigma_t^2}{2}+C\alpha_t^2\delta_t^2\right].$$

By the joint convexity of $F(\mathbf{w},\nu)$, we have

$$\alpha_t(F(\mathbf{w}_t,\nu_t)-F(\mathbf{w}_*,\nu_*)\leq\alpha_t(\nabla_1 F(\mathbf{w}_t,\nu_t)^\top(\mathbf{w}_t-\mathbf{w}_*)+\nabla_2 F(\mathbf{w}_t,\nu_t)^\top(\nu_t-\nu_*)).$$

Combining the last two inequalities and summing over $t=1,\ldots,T$, we have

$$\mathbb{E}\left[\sum_{t=1}^{T}\alpha_t(F(\mathbf{w}_t,\nu_t)-F(\mathbf{w}_*,\nu_*))\right]$$
$$\leq\frac{1}{2\eta}\|\mathbf{w}_1-\mathbf{w}\|_2^2+D_\varphi(\nu_*,\nu_0)+\mathbb{E}\left[\sum_{t=1}^{T}\frac{\eta\alpha_t^2\sigma_t^2}{2}+\sum_{t=1}^{T}C\alpha_t^2\delta_t^2\right].$$

□

We present two corollaries of the above theorem.

Corollary 5.2 *Suppose Assumption 5.17 holds. Let $\beta_t=1,\eta_t=\eta\alpha_t$, $\alpha_t=\frac{\alpha}{\sqrt{T}}<\rho e^{-v_{t-1}}$ for some constant $\rho>0$, then SCENT guarantees that*

$$\mathbb{E}\left[(F_1(\bar{\mathbf{w}}_T)-F_1(\mathbf{w}_*))\right]\leq\frac{D_0}{\alpha\sqrt{T}}+\frac{\alpha V}{\sqrt{T}}.$$

where $\bar{\mathbf{w}}_T=\frac{\sum_{t=1}^{T}\mathbf{w}_t}{T}$, $D_0=\frac{1}{2\eta}\|\mathbf{w}_1-\mathbf{w}_*\|_2^2+D_\varphi(\nu_*,\nu_0)$ and

$$V=\mathbb{E}\left[\frac{\eta\sum_{t=1}^{T}\sigma_t^2}{2T}+\frac{\sum_{t=1}^{T}C\delta_t^2}{T}\right].$$

Proof. Plugging $\alpha_t = \alpha/\sqrt{T}$ into Theorem 5.14, we have

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}(F(\mathbf{w}_t, \nu_t) - F(\mathbf{w}_*, \nu_*))\right] \leq \frac{D_0}{\alpha\sqrt{T}} + \frac{\alpha V}{\sqrt{T}}.$$

Using $F_1(\mathbf{w}) = \min_\nu F(\mathbf{w}, \nu), F_1(\mathbf{w}_*) = F(\mathbf{w}_*, \nu_*)$ and the Jensen inequality, we can finish the proof. □

Why it matters

Since δ_t, σ_t are finite, the above result implies a convergence rate of $O(1/\sqrt{T})$ for SCENT.

Corollary 5.3 *Suppose Assumption 5.17 holds. Let* $\beta_t = 1, \eta_t = \eta\alpha_t$, $\alpha_t = \frac{\alpha e^{-\nu_{t-1}}}{\sqrt{T}}$, *if* $\frac{1}{T}\sum_{t=1}^{T} e^{-\nu_{t-1}} \geq S$ *almost surely, then SCENT guarantees that*

$$\mathbb{E}\left[F_1(\hat{\mathbf{w}}_T) - F_1(\mathbf{w}_*)\right] \leq \frac{D_0}{\alpha\sqrt{T}S} + \frac{\alpha\bar{V}}{\sqrt{T}S}.$$

where $\hat{\mathbf{w}}_T = \frac{\sum_t \alpha_t \mathbf{w}_t}{\sum_{t=1}^{T}\alpha_t}$ *and*

$$\bar{V} = \mathbb{E}\left[\frac{\eta\sum_{t=1}^{T} e^{-2\nu_{t-1}}\sigma_t^2}{2T} + \frac{\sum_{t=1}^{T} C e^{-2\nu_{t-1}}\delta_t^2}{T}\right].$$

Proof. Let $\hat{\alpha}_t = \frac{\alpha_t}{\sum_{t=1}^{T}\alpha_t}$. From Theorem 5.14, we have

$$\begin{aligned}
&\mathbb{E}\left[\sum_{t=1}^{T}\alpha_t\sum_{t=1}^{T}\hat{\alpha}_t(F(\mathbf{w}_t, \nu_t) - F(\mathbf{w}_*, \nu_*))\right] \\
&\leq \frac{1}{2\eta}\|\mathbf{w}_1 - \mathbf{w}\|_2^2 + D_\varphi(\nu_*, \nu_0) + \mathbb{E}\left[\sum_{t=1}^{T}\frac{\eta\alpha_t^2\sigma_t^2}{2} + \sum_{t=1}^{T} C\alpha_t^2\delta_t^2\right].
\end{aligned}$$

Since $\sum_{t=1}^{T}\alpha_t = \sum_{t=1}^{T}\frac{\alpha e^{-\nu_{t-1}}}{\sqrt{T}} \geq \alpha\sqrt{T}S$, then

$$\mathbb{E}\left[\sum_{t=1}^{T}\hat{\alpha}_t(F(\mathbf{w}_t, \nu_t) - F(\mathbf{w}_*, \nu_*))\right] \leq \frac{\frac{1}{2\eta}\|\mathbf{w}_1 - \mathbf{w}\|_2^2 + D_\varphi(\nu_*, \nu_0)}{\alpha\sqrt{T}S} + \frac{\alpha\bar{V}}{\sqrt{T}S}.$$

Applying the joint convexity of $F(\mathbf{w}, \nu)$ and $F_1 = \min_\nu F(\mathbf{w}, \nu)$, we can finish the proof. □

Why it matters

Under the stated setting, SCENT reduces to SCGD with $\gamma_t = \frac{\alpha}{\sqrt{T}+\alpha}$. Since S can be lower bounded by a constant, the above corollary implies $O(1/\sqrt{T})$ convergence rate for SCGD to minimize log-E-Exp.

Analysis of the Variance Terms

Since the final convergence bound depends on the variance terms σ_t^2, δ_t^2, we would like to provide further analysis on them.

Let us introduce some notations:

$$z(\mathbf{w};\zeta) = e^{s(\mathbf{w};\zeta)},\ \mu(\mathbf{w}) = \log \mathbb{E}_\zeta e^{s(\mathbf{w};\zeta)}, \tag{5.88}$$

$$m_t = \mathbb{E}_\zeta e^{s(\mathbf{w}_t;\zeta)},\ \mu_t = \mu(\mathbf{w}_t) = \log m_t. \tag{5.89}$$

For the analysis, we make two reasonable assumptions.

Assumption 5.18. *Assume there exist constants κ, σ'^2 such that (i) $\mathbb{E}\left[\frac{\mathbb{E}[z(\mathbf{w};\zeta)^2]}{(\mathbb{E}[z(\mathbf{w};\zeta)])^2}\right] \leq \kappa$ for all $\mathbf{w}$; (ii) $\mathbb{E}\|e^{s(\mathbf{w}_t;\zeta')-\mu_t}\nabla s(\mathbf{w}_t;\zeta')\|^2 \leq \sigma'^2$ for all t;*

Critical: These assumptions are necessary. In next section, we show that the dependence on κ is unavoidable. The second assumption is the standard bounded stochastic gradient assumption for optimizing $F_1(\mathbf{w})$.

Lemma 5.32 (Dual Variance Term) *Under Assumption 5.18, we have*

$$\delta_t^2 \leq 2(\kappa-1)m_t\Big(F(\mathbf{w}_t,\nu_{t-1}) - F(\mathbf{w}_*,\nu_*) + 1\Big). \tag{5.90}$$

Why it matters

When $F(\mathbf{w}_t,\nu_{t-1}) - F(\mathbf{w}_*,\nu_*) \to 0$, the variance term in the convergence bound caused by the stochastic update of ν_t will be dominated by $2(\kappa-1)m_t$. Large m_t can be mitigated by choosing small α_t.

Proof. Recall that

$$\delta_t^2 = \mathbb{E}_{\zeta_t}\left[e^{-\nu_{t-1}}\big(z(\mathbf{w}_t;\zeta_t) - m_t\big)^2\right]$$

By Assumption 5.18(i),

$$\mathrm{Var}(z(\mathbf{w}_t;\zeta)) \leq (\kappa-1)m_t^2.$$

Hence

$$\delta_t^2 = e^{-\nu_{t-1}} \operatorname{Var}(z(\mathbf{w}_t; \zeta)) \leq (\kappa - 1) e^{-\nu_{t-1}} m_t^2 = (\kappa - 1) m_t \cdot (m_t e^{-\nu_{t-1}}).$$

Let $\tilde{r}_{t-1} \coloneqq m_t e^{-\nu_{t-1}}$. By the definition:

$$F(\mathbf{w}_t, \nu_{t-1}) = \mathbb{E} e^{s(\mathbf{w}_t; \zeta) - \nu_{t-1}} + \nu_{t-1} = \tilde{r}_{t-1} + \nu_{t-1}.$$

Since $\tilde{r}_{t-1} = e^{\log m_t - \nu_{t-1}}$, we have

$$F(\mathbf{w}_t, \nu_{t-1}) - (1 + \mu_t) = \tilde{r}_{t-1} + \nu_{t-1} - (1 + \log m_t) = \tilde{r}_{t-1} - \log \tilde{r}_{t-1} - 1.$$

Using $r \leq 2(r - \log r)$ for all $r > 0$ yields

$$\tilde{r}_{t-1} \leq 2\big(F(\mathbf{w}_t, \nu_{t-1}) - (1 + \mu_t) + 1\big).$$

Since $\mathbf{w}_*$ minimizes $\mu(\mathbf{w})$, we have $\mu_t = \mu(\mathbf{w}_t) \geq \mu(\mathbf{w}_*)$ and thus $(1 + \mu_t) \geq (1 + \mu(\mathbf{w}_*)) = F(\mathbf{w}_*, \nu_*)$, implying

$$F(\mathbf{w}_t, \nu_{t-1}) - (1 + \mu_t) \leq F(\mathbf{w}_t, \nu_{t-1}) - F(\mathbf{w}_*, \nu_*).$$

As a result, we have

$$\tilde{r}_{t-1} \leq 2\big(F(\mathbf{w}_t, \nu_{t-1}) - F(\mathbf{w}_*, \nu_*) + 1\big). \tag{5.91}$$

Combining this with the bound of δ_t^2, we complete the proof. □

Lemma 5.33 (Primal Variance Term) *Under Assumption 5.18, we have*

$$\sigma_t^2 \leq 4\sigma'^2 \big(F(\mathbf{w}_t, \nu_t) - F(\mathbf{w}_*, \nu_*) + 1\big)^2.$$

Why it matters

When $F(\mathbf{w}_t, \nu_t) - F(\mathbf{w}_*, \nu_*) \to 0$, the variance term in the convergence bound caused by the stochastic update of $\mathbf{w}_t$ will be dominated by $O(\sigma'^2)$.

Proof.

$$\begin{aligned}
\sigma_t^2 &= \mathbb{E}_{\zeta_t'} \| \exp(s(\mathbf{w}_t; \zeta_t') - \nu_t) \nabla s(\mathbf{w}_t; \zeta_t') \|_2^2], \\
&= \mathbb{E}_{\zeta_t'} [e^{2(\mu_t - \nu_t)} \| \exp(s(\mathbf{w}_t; \zeta_t') - \mu_t) \nabla s(\mathbf{w}_t; \zeta_t') \|_2^2] \leq r_t^2 \sigma'^2,
\end{aligned}$$

where $r_t = e^{\mu_t - \nu_t}$. Similar to (5.91), we have show that

$$r_t \leq 2\big(F(\mathbf{w}_t, \nu_t) - F(\mathbf{w}_*, \nu_*) + 1\big).$$

Hence,

$$\sigma_t^2 \leq 4\sigma'^2 \big(F(\mathbf{w}_t, \nu_t) - F(\mathbf{w}_*, \nu_*) + 1\big)^2.$$

□

Algorithm 22 The SCENT Algorithm for solving CERM

1: Initialize $\mathbf{w}_1, \boldsymbol{\nu}_0$, step sizes η_t and α_t, $\varphi(\nu) = e^{-\nu}$.
2: **for** $t = 1 \ldots, T-1$ **do**
3: Sample $\mathcal{B}_t \subset \{1, \ldots, n\}$ with $|\mathcal{B}_t| = B$
4: **for** each $i \in \mathcal{B}_t$ **do**
5: Sample $\zeta_{i,t}, \zeta'_{i,t} \sim \mathbb{P}_i$
6: Update $\nu_{i,t} = \arg\min_\nu \exp(s_i(\mathbf{w}_t; \zeta_{i,t}) - \nu) + \nu + \frac{1}{\alpha_t} D_\varphi(\nu, \nu_{i,t-1})$
7: **end for**
8: Compute $\mathbf{z}_t = \frac{1}{B} \sum_{i \in \mathcal{B}_t} \exp(s_i(\mathbf{w}_t; \zeta'_{i,t}) - \nu_{i,t}) \nabla s_i(\mathbf{w}_t; \zeta'_{i,t})$
9: Compute $\mathbf{v}_t = (1 - \beta_t)\mathbf{v}_{t-1} + \beta_t \mathbf{z}_t$
10: Update $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{v}_t$
11: **end for**

5.5.2.2 Compositional Entropic Risk Minimization

In this section, we extend the results to solving compositional entropic risk minimization (CERM):

$$\min_{\mathbf{w}} F_1(\mathbf{w}) := \frac{1}{n} \sum_{i=1}^{n} \log\left(\mathbb{E}_{\zeta \sim \mathbb{P}_i} \exp(s_i(\mathbf{w}; \zeta))\right)$$

via its equivalent min-min formulation:

$$\min_{\mathbf{w}} \min_{\boldsymbol{\nu}} F(\mathbf{w}, \boldsymbol{\nu}) := \frac{1}{n} \sum_{i=1}^{n} \{\mathbb{E}_{\zeta \sim \mathbb{P}_i} \exp(s_i(\mathbf{w}; \zeta) - \nu_i) + \nu_i\}.$$

The difference from Log-E-Exp is that there are multiple $\nu_i, i = 1, \ldots, n$, which needs to be updated using stochastic block coordinate method. The technique has been used in algorithms presented in previous sections of this chapter.

We present an extension of SCENT to solving CERM in Algorithm 22. The major change lies at the stochastic block coordinate update of $\boldsymbol{\nu}$ in Step 5. This extension is analogous to SOX for FCCO, employing stochastic block-coordinate updates for the inner estimators. Indeed, SOX applied to CERM can be recovered as a special case of SCENT by choosing the coordinate-wise step size $\alpha_{t,i} = \frac{\gamma_t}{1-\gamma_t} e^{-\nu_{i,t-1}}$, using an argument similar to (5.83).

Convergence analysis for convex problems

Let us define some notations:

$$\Phi_i(\mathbf{w}_t, \nu_i; \zeta) = \exp(s_i(\mathbf{w}_t; \zeta_t) - \nu_i) + \nu_i$$
$$F_i(\mathbf{w}_t, \nu_i) = \mathbb{E}_{\zeta \sim \mathbb{P}_i}[\Phi_i(\mathbf{w}_t, \nu_i; \zeta)]$$
$$(\mathbf{w}_*, \boldsymbol{\nu}_*) = \arg\min_{\mathbf{w}, \boldsymbol{\nu}} F(\mathbf{w}, \boldsymbol{\nu}).$$

Similar as before, $\nu_{i,*} = \log[\mathbb{E}_{\zeta\sim\mathbb{P}_i}\exp(s_i(\mathbf{w}_*;\zeta))]$. Since we deal with stochastic block coordinate update, we introduce a virtual sequence $\bar{\boldsymbol{\nu}}_t$, where

$$\bar{\nu}_{i,t} = \arg\min_{\nu} \exp(s_i(\mathbf{w}_t;\zeta_{i,t}) - \nu) + \nu + \frac{1}{\alpha_t} D_\varphi(\nu, \nu_{i,t-1}), \forall i$$

Following Lemma 5.26, we have

$$e^{\bar{\nu}_{i,t}} = \frac{1}{1+\alpha_t e^{\nu_{i,t-1}}} e^{\nu_{i,t-1}} + \frac{\alpha_t e^{\nu_{i,t-1}}}{1+\alpha_t e^{\nu_{i,t-1}}} \exp(s_i(\mathbf{w}_t;\zeta_t)), \forall i.$$

Assumption 5.19. *Assume that the following conditions hold:*

(i) $s_i(\mathbf{w};\zeta)$ *is convex;*
(ii) the loss function is bounded such that $s_i(\mathbf{w};\zeta) \in [c_0, c_1], \forall \mathbf{w}, \zeta, i.$
(iii) there exists G *such that* $\mathbb{E}_\zeta \|\nabla s_i(\mathbf{w}_t,\zeta)\|_2^2] \le G^2, \forall t, i$

Define $\sigma_{i,t}, \delta_{i,t}$ as

$$\sigma_{i,t}^2 := \mathbb{E}_{\zeta'_{i,t}\sim\mathbb{P}_i} \| \exp(s_i(\mathbf{w}_t;\zeta'_{i,t}) - \nu_{i,t}) \nabla s_i(\mathbf{w}_t;\zeta'_{i,t})\|_2^2], \forall i, t,$$
$$\delta_{i,t}^2 := \mathbb{E}_{\zeta_{i,t}\sim\mathbb{P}_i}[e^{-\nu_{i,t-1}} | e^{s_i(\mathbf{w}_t;\zeta_{i,t})} - \mathbb{E}_{\zeta_i\sim\mathbb{P}_i}[e^{s_i(\mathbf{w}_t;\zeta_i)}]|^2], \forall i, t.$$

Similar to Lemma 5.27, the following lemma can be proved.

Lemma 5.34 *Under Assumption 5.19, if* $\boldsymbol{\nu}_0 \in [c_0, c_1]$ *then* $\boldsymbol{\nu}_t \in [c_0, c_1], \forall t.$

Similar to Lemma 5.28, we have the following lemma regarding one-step update of $\mathbf{w}_t$.

Lemma 5.35 *Under Assumption (5.19) and* $\beta_t = 1$*, we have*

$$\mathbb{E}[\eta_t \nabla_1 F(\mathbf{w}_t, \bar{\boldsymbol{\nu}}_t)^\top (\mathbf{w}_t - \mathbf{w}_*)] \le \mathbb{E}\left[\frac{1}{2}\|\mathbf{w}_t - \mathbf{w}_*\|_2^2 - \frac{1}{2}\|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2^2\right] + \frac{\eta_t^2 \sigma_t^2}{2},$$

where $\sigma_t^2 = \frac{1}{n}\sum_{i=1}^n \sigma_{i,t}^2$.

Proof. We first bound $\mathbb{E}_t[\|\mathbf{z}_t\|_2^2 \mid \mathcal{F}_{t-1}]$, where $\mathbb{E}_t$ denotes the expectation over randomness in t-th iteration given $\mathbf{w}_t, \nu_{t-1}$.

$$\mathbb{E}_t[\|\mathbf{z}_t\|_2^2] = \mathbb{E}_t\left[\left\|\frac{1}{B}\sum_{i\in\mathcal{B}_t} \exp(s_i(\mathbf{w}_t;\zeta'_{i,t}) - \nu_{i,t})\nabla s_i(\mathbf{w}_t;\zeta'_{i,t})\right\|_2^2\right]$$
$$= \mathbb{E}_{\mathcal{B}_t,\zeta_t}\mathbb{E}_{\zeta'_t|\mathcal{B}_t,\zeta_t}\left[\left\|\frac{1}{B}\sum_{i\in\mathcal{B}_t} \exp(s_i(\mathbf{w}_t;\zeta'_{i,t}) - \nu_{i,t})\nabla s_i(\mathbf{w}_t;\zeta'_{i,t})\right\|_2^2\right]$$
$$\le \mathbb{E}_{\mathcal{B}_t,\zeta_t}\left[\frac{1}{B}\sum_{i\in\mathcal{B}_t}\sigma_{i,t}^2\right] = \frac{1}{n}\sum_{i=1}^n \sigma_{i,t}^2.$$

Since $\bar{\nu}_{i,t} = \nu_{i,t}, \forall i \in \mathcal{B}_t$, we have

$$\mathbb{E}_t[\mathbf{z}_t] = \mathbb{E}_{\zeta'_t,\zeta_t,\mathcal{B}_t}\left[\frac{1}{B}\sum_{i\in\mathcal{B}_t}\nabla\Phi_i(\mathbf{w}_t,\bar{\nu}_{i,t};\zeta'_{i,t})\right] = \nabla_1 F(\mathbf{w}_t,\bar{\boldsymbol{\nu}}_t).$$

Then following Lemma 3.3, we can finish the proof. □

Next, we analyze the update of $\bar{\nu}_t$.

Lemma 5.36 *Under Assumption (5.19) (ii) and* $\alpha_t \le \min_i \rho e^{-\nu_{i,t-1}}$*, we have*

$$\mathbb{E}[\alpha_t\nabla_2 F(\mathbf{w}_t,\bar{\boldsymbol{\nu}}_t)^\top(\bar{\boldsymbol{\nu}}_t-\boldsymbol{\nu}_*)] \le \frac{1}{B}\mathbb{E}\left[D_\varphi(\boldsymbol{\nu}_*,\boldsymbol{\nu}_{t-1}) - D_\varphi(\boldsymbol{\nu}_*,\boldsymbol{\nu}_t)\right] + C\alpha_t^2\delta_t^2.$$

where $D_\varphi(\boldsymbol{\nu}_*,\boldsymbol{\nu}_t) = \sum_{i=1}^n D_\varphi(\nu_{i,*},\nu_{i,t})$ *and* $\delta_t^2 = \frac{1}{n}\sum_{i=1}^n \delta_{i,t}^2$.

Proof. By applying Lemma 5.30 and Lemma 5.29 for each coordinate of $\bar{\nu}_{i,t}$, we have

$$\mathbb{E}[\alpha_t\nabla_2 F_i(\mathbf{w}_t,\bar{\nu}_{i,t})^\top(\bar{\nu}_{i,t}-\nu_{i,*})] \le D_\varphi(\nu_{i,*},\nu_{i,t-1}) - D_\varphi(\nu_{i,*},\bar{\nu}_{i,t}) + C\alpha_t^2\delta_{i,t}^2, \forall i.$$

Averaging the above inequality over $i = 1,\ldots,n$, we have

$$\mathbb{E}[\alpha_t\nabla_2 F(\mathbf{w}_t,\bar{\boldsymbol{\nu}}_t)^\top(\bar{\boldsymbol{\nu}}_t-\boldsymbol{\nu}_*)] \le \frac{1}{n}\sum_{i=1}^n\left(D_\varphi(\nu_{i,*},\nu_{i,t-1}) - D_\varphi(\nu_{i,*},\bar{\nu}_{i,t})\right) + C\alpha_t\delta_t^2. \tag{5.92}$$

Due to the randomness of $\mathcal{B}_t$, we have

$$\mathbb{E}[D_\varphi(\nu_{i,*},\nu_{i,t})] = \mathbb{E}\left[(1-\frac{B}{n})D_\varphi(\nu_{i,*},\nu_{i,t-1}) + \frac{B}{n}D_\varphi(\nu_{i,*},\bar{\nu}_{i,t})\right], \forall i.$$

Hence

$$\begin{aligned}
&\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^n\left(D_\varphi(\nu_{i,*},\nu_{i,t-1}) - D_\varphi(\nu_{i,*},\bar{\nu}_{i,t})\right)\right]\\
&= \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^n\left(D_\varphi(\nu_{i,*},\nu_{i,t-1}) - \frac{n}{B}D_\varphi(\nu_{i,*},\nu_{i,t}) + (\frac{n}{B}-1)D_\varphi(\nu_{i,*},\nu_{i,t-1})\right)\right]\\
&= \frac{1}{B}\mathbb{E}\left[\sum_{i=1}^n(D_\varphi(\nu_{i,*},\nu_{i,t-1}) - D_\varphi(\nu_{i,*},\nu_{i,t}))\right].
\end{aligned}$$

Combining this with (5.92), we finish the proof. □

Finally, we state the convergence result of SCENT in the following theorem.

Theorem 5.15 *Suppose Assumption 5.19 holds. Let* $\beta_t = 1, \eta_t = \eta\alpha_t$*, and* $\alpha_t = \frac{\alpha}{\sqrt{T}} < \rho\min_i e^{-v_{i,t-1}}$*, then SCENT guarantees that*

$$\mathbb{E}\left[(F_1(\bar{\mathbf{w}}_T) - F_1(\mathbf{w}_*))\right] \leq \frac{1}{2\eta\alpha\sqrt{T}}\|\mathbf{w}_1 - \mathbf{w}_*\|_2^2 + \frac{D_\varphi(\boldsymbol{\nu}_*, \boldsymbol{\nu}_0)}{\alpha B\sqrt{T}} + \frac{\alpha V}{\sqrt{T}}.$$

where $\bar{\mathbf{w}}_T = \frac{\sum_{t=1}^T \mathbf{w}_t}{T}$, *and* $V = \mathbb{E}\left[\frac{\eta \sum_{t=1}^T \sigma_t^2}{2T} + \frac{\sum_{t=1}^T C\delta_t^2}{T}\right]$.

Why it matters

In order to achieve an ϵ-optimal solution, the above convergence bound implies the following complexity:

$$T = O\left(\frac{\|\mathbf{w}_1 - \mathbf{w}_*\|_2^4}{\eta^2\alpha^2\epsilon^2} + \frac{D_\varphi(\boldsymbol{\nu}_*, \boldsymbol{\nu}_0)^2}{\alpha^2 B^2\epsilon^2} + \frac{\alpha^2 V^2}{\epsilon^2}\right).$$

For simplicity of discussion, let us consider a setting of η such that the first term matches the second term. As a result, the complexity becomes:

$$T = O\left(\frac{D_\varphi(\boldsymbol{\nu}_*, \boldsymbol{\nu}_0)^2}{\alpha^2 B^2\epsilon^2} + \frac{\alpha^2 V^2}{\epsilon^2}\right).$$

Insight 1: Since σ_t, δ_t are finite, and $D_\varphi(\boldsymbol{\nu}_*, \boldsymbol{\nu}_0) = O(n)$, if $\alpha \propto \sqrt{n/B}$, the above result implies an iteration complexity of $O(\frac{n}{B\epsilon^2})$ for SCENT.
Insight 2: When the loss $s_i(\mathbf{w}_t; \zeta) \geq 0$ is large, the term $e^{-\nu_{i,t-1}}$ becomes very small, suggesting that the step size parameter α should be chosen small so as to mitigate the large variance term δ_t. In contrast, when the loss $s_i(\mathbf{w}_t; \zeta) < 0$ is small, the term $e^{-\nu_{i,t-1}}$ can become large, allowing α to be set relatively larger, which helps offset the large distance measure $D_\varphi(\boldsymbol{\nu}_*, \boldsymbol{\nu}_0)$.

Proof. Since $\eta_t = \eta\alpha_t$, from Lemma 5.35, we obtain

$$\mathbb{E}[\alpha_t \nabla_1 F(\mathbf{w}_t, \bar{\boldsymbol{\nu}}_t)^\top (\mathbf{w}_t - \mathbf{w}_*)] \leq \mathbb{E}\left[\frac{1}{2\eta}\|\mathbf{w}_t - \mathbf{w}\|_2^2 - \frac{1}{2\eta}\|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2^2 + \frac{\eta\alpha_t^2\sigma_t^2}{2}\right].$$

Adding this to the inequality in Lemma 5.36, we have

$$\begin{aligned}
&\mathbb{E}[\alpha_t(\nabla_1 F(\mathbf{w}_t, \bar{\boldsymbol{\nu}}_t)^\top (\mathbf{w}_t - \mathbf{w}_*) + \nabla_2 F(\mathbf{w}_t, \bar{\boldsymbol{\nu}}_t)^\top (\bar{\boldsymbol{\nu}}_t - \boldsymbol{\nu}_*))] \\
&\leq \mathbb{E}\left[\frac{1}{2\eta}\|\mathbf{w}_t - \mathbf{w}\|_2^2 - \frac{1}{2\eta}\|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2^2 + \frac{1}{B}D_\varphi(\boldsymbol{\nu}_*, \boldsymbol{\nu}_{t-1}) - \frac{1}{B}D_\varphi(\boldsymbol{\nu}_*, \boldsymbol{\nu}_t)\right] \\
&+ \mathbb{E}\left[\frac{\eta\alpha_t^2\sigma_t^2}{2} + C\alpha_t^2\delta_t^2\right].
\end{aligned}$$

By the joint convexity of $F(\mathbf{w}, \nu)$, we have

$$\alpha_t(F(\mathbf{w}_t, \bar{\boldsymbol{\nu}}_t) - F(\mathbf{w}_*, \boldsymbol{\nu}_*) \leq \alpha_t(\nabla_1 F(\mathbf{w}_t, \bar{\boldsymbol{\nu}}_t)^\top (\mathbf{w}_t - \mathbf{w}_*) + \nabla_2 F(\mathbf{w}_t, \bar{\boldsymbol{\nu}}_t)^\top (\bar{\boldsymbol{\nu}}_t - \boldsymbol{\nu}_*)).$$

Combining the last two inequalities and summing over $t = 1, \ldots, T$, we have

$$\mathbb{E}\left[\sum_{t=1}^{T}\alpha_t(F(\mathbf{w}_t,\bar{\boldsymbol{\nu}}_t)-F(\mathbf{w}_*,\boldsymbol{\nu}_*))\right]$$
$$\leq \frac{1}{2\eta}\|\mathbf{w}_1-\mathbf{w}\|_2^2+\frac{1}{B}D_\varphi(\boldsymbol{\nu}_*,\boldsymbol{\nu}_0)+\mathbb{E}\left[\sum_{t=1}^{T}\frac{\eta\alpha_t^2\sigma_t^2}{2}+\sum_{t=1}^{T}C\alpha_t^2\delta_t^2\right].$$

Since $F_1(\mathbf{w}_*)=F(\mathbf{w}_*,\boldsymbol{\nu}_*)$, and $F_1(\mathbf{w}_t)\leq F(\mathbf{w}_t,\bar{\boldsymbol{\nu}}_t)$, we have

$$\mathbb{E}\left[\sum_{t=1}^{T}\alpha_t(F_1(\mathbf{w}_t)-F_1(\mathbf{w}_*))\right]$$
$$\leq \frac{1}{2\eta}\|\mathbf{w}_1-\mathbf{w}\|_2^2+\frac{1}{B}D_\varphi(\boldsymbol{\nu}_*,\boldsymbol{\nu}_0)+\mathbb{E}\left[\sum_{t=1}^{T}\frac{\eta\alpha_t^2\sigma_t^2}{2}+\sum_{t=1}^{T}C\alpha_t^2\delta_t^2\right].$$

Plugging the value of α_t, we finish the proof.

□

5.5.2.3 Why SCENT is better than ASGD?

In this section, we provide theoretical insight into why SCENT outperforms ASGD for entropic risk minimization. The key distinction between the two methods lies in their updates of the dual variable ν: SCENT employs a stochastic proximal mirror descent (SPMD) update, whereas ASGD relies on a standard SGD update. Accordingly, our analysis focuses exclusively on the ν-update while keeping $\mathbf{w}$ fixed. In particular, we consider the following problem:

$$\min_{\nu} F(\nu) := \mathbb{E}_\zeta e^{s(\zeta)-\nu}+\nu, \tag{5.93}$$

where we omit $\mathbf{w}$ in $s(\zeta)$.

Recall the definitions $z \coloneqq e^{s(\zeta)}$, $m \coloneqq \mathbb{E}[z]$, $r(\nu) \coloneqq me^{-\nu}=e^{\nu_*-\nu}$ as used previously, and the facts $\nu_*=\arg\min_\nu F(\nu)=\log m, F(\nu_*)=me^{-\nu_*}+\nu_*=1+\nu_*$. Recall the SPMD update:

$$e^{\nu_t}=\frac{1}{1+\alpha_t e^{\nu_{t-1}}}e^{\nu_{t-1}}+\frac{\alpha_t e^{\nu_{t-1}}}{1+\alpha_t e^{\nu_{t-1}}}e^{s(\zeta_t)}.$$

Let us define an important quantity to characterize the difficulty of the problem:

$$\kappa=\frac{\mathbb{E}[z^2]}{(\mathbb{E}[z])^2},$$

which is known as second-order moment ratio. Larger κ indicates heavier tails or higher variability relative to the mean.

A Clean Bound of SPMD

The optimality gap can be written as

$$F(\nu) - F(\nu_*) = me^{-\nu} + \nu - (1 + \nu_*) = r(\nu) - \log r(\nu) - 1. \tag{5.94}$$

We assume $s(\zeta) \in [c_0, c_1]$ and without loss of generality we assume $c_1 \leq 0$. If not, we can define $s'(\zeta) = s(\zeta) - c_1, z' = e^{s'(\zeta)}$ and $F'(\nu') = \mathbb{E}[z'e^{-\nu'}] + \nu'$. Then $F(\nu) - F(\nu_*) = F'(\nu') - \min F'(\nu)$ if $\nu = \nu' - c_1$.

Lemma 5.37 (Self-bounding inequality) *For all $r > 0$,*

$$r \;\leq\; 2\,(r - \log r). \tag{5.95}$$

Equivalently, for all $\nu \in \mathbb{R}$,

$$r(\nu) \leq 2\big(F(\nu) - F(\nu_*) + 1\big). \tag{5.96}$$

Proof. If $0 < r \leq 2$, then $r \leq 2 \leq 2(r - \log r)$ since $r - \log r \geq 1$ for all $r > 0$. If $r \geq 2$, then $\log r \leq r/2$, hence $r - \log r \geq r/2$, i.e. $r \leq 2(r - \log r)$. Substituting $r = r(\nu)$ and using (5.94) yields (5.96). □

Theorem 5.16 *Suppose $s(\zeta) \in [c_0, c_1] \leq 0$ holds. By setting $\alpha_t = \sqrt{\frac{D_\varphi(\nu_*, \nu_0)m}{2CT\mathrm{Var}(z)}} \leq \min(\frac{m}{4C\mathrm{Var}(z)}, \rho)$ for sufficiently large T, SPMD guarantees that*

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[F(\nu_t) - F(\nu_*)] \leq 4\sqrt{2}\sqrt{\frac{C\,(\kappa - 1)\,\big(1 - r_0 + r_0 \log r_0\big)}{T}} + \frac{F(\nu_0) - F(\nu_*)}{T}. \tag{5.97}$$

where $C = (1 + \rho)(1 + c_1 - c_0)$, and $r_0 = r(\nu_0) = e^{\nu_* - \nu_0}$.

Why it matters

When $\nu_0 \gg \nu_*$ (over-estimation), then $1 - r_0 + r_0 \log r_0 = O(1)$, the dominating term becomes $O(\sqrt{\frac{\kappa}{T}})$. This upper bound characterizes the intrinsic complexity of SPMD, which depends on the second-order moment ratio κ. If $s(\zeta)) \sim \mathcal{N}(\mu_s, \sigma_s^2)$, then $\kappa = e^{\sigma_s^2}$, which does not depend on the exponential of the mean μ_s but rather $e^{\sigma_s^2}$.

Proof. From Lemma 5.31, we obtain the SPMD averaged bound

$$\bar{G}_T \;\coloneqq\; \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[F(\nu_t) - F(\nu_*)] \;\leq\; \frac{D_\varphi(\nu_*, \nu_0)}{\alpha T} \;+\; C\,\alpha\,V, \tag{5.98}$$

where

$$V \coloneqq \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[\delta_t^2], \qquad \delta_t^2 = \mathbb{E}\big[e^{-\nu_{t-1}}(z_t - m)^2\big] = e^{-\nu_{t-1}}\operatorname{Var}(z).$$

Since $e^{-\nu_{t-1}} = r(\nu_{t-1})/m$, we can rewrite

$$V = \frac{\operatorname{Var}(z)}{m} \cdot \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[r(\nu_{t-1})]. \tag{5.99}$$

By Lemma 5.37,

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[r(\nu_{t-1})] &\le \frac{2}{T}\sum_{t=1}^{T}\mathbb{E}[F(\nu_{t-1}) - F(\nu_*) + 1] \\ &= 2\left(1 + \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[F(\nu_{t-1}) - F(\nu_*)]\right).\end{aligned}$$

Next, observe the index shift:

$$\begin{aligned}&\sum_{t=1}^{T}\mathbb{E}[F(\nu_{t-1}) - F(\nu_*)] = \mathbb{E}[F(\nu_0) - F(\nu_*)] + \sum_{t=1}^{T-1}\mathbb{E}[F(\nu_t) - F(\nu_*)] \\ &\le \mathbb{E}[F(\nu_0) - F(\nu_*)] + \sum_{t=1}^{T}\mathbb{E}[F(\nu_t) - F(\nu_*)].\end{aligned}$$

Dividing by T yields

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[F(\nu_{t-1}) - F(\nu_*)] \le \frac{\mathbb{E}[F(\nu_0) - F(\nu_*)]}{T} + \bar{G}_T. \tag{5.100}$$

Combining this with (5.99) we have

$$V \le \frac{2\operatorname{Var}(z)}{m}\left(1 + \bar{G}_T + \frac{\mathbb{E}[F(\nu_0) - F(\nu_*)]}{T}\right). \tag{5.101}$$

Plugging (5.101) into (5.98) yields

$$\bar{G}_T \le \frac{D_\varphi(\nu_*, \nu_0)}{\alpha T} + \frac{2C\alpha\operatorname{Var}(z)}{m}\left(1 + \bar{G}_T + \frac{\mathbb{E}[F(\nu_0) - F(\nu_*)]}{T}\right).$$

If $\alpha \le \frac{m}{4C\operatorname{Var}(z)}$, then $\frac{2C\alpha\operatorname{Var}(z)}{m} \le \frac{1}{2}$, and therefore

$$\bar{G}_T \le \frac{2D_\varphi(\nu_*,\nu_0)}{\alpha T} + \frac{4C\alpha\,\mathrm{Var}(z)}{m}\left(1+\frac{\mathbb{E}[F(\nu_0)-F(\nu_*)]}{T}\right)$$
$$\le \frac{2D_\varphi(\nu_*,\nu_0)}{\alpha T} + \frac{4C\alpha\,\mathrm{Var}(z)}{m} + \frac{F(\nu_0)-F(\nu_*)}{T}.$$

Optimizing the right-hand side over α (assuming T is large enough) gives:

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[F(\nu_t)-F(\nu_*)] \le 4\sqrt{2}\sqrt{\frac{C\,D_\varphi(\nu_*,\nu_0)\mathrm{Var}(z)}{mT}} + \frac{F(\nu_0)-F(\nu_*)}{T}.$$

With $r_0 \coloneqq r(\nu_0) = e^{\nu_*-\nu_0}$,

$$D_\varphi(\nu_*,\nu_0) = e^{-\nu_*} - e^{-\nu_0} + e^{-\nu_0}(\nu_*-\nu_0) = \frac{1}{m}\big(1-r_0+r_0\log r_0\big).$$

Since $\mathrm{Var}(z)/m^2 = \kappa - 1$, thus the convergence upper bound becomes

$$4\sqrt{2}\sqrt{\frac{C\,(\kappa-1)\big(1-r_0+r_0\log r_0\big)}{T}} + \frac{F(\nu_0)-F(\nu_*)}{T}.$$

□

Comparison with SGD.

Benefit under the noise setting

In order to control the variance, we consider projected SGD. Let $\Pi_{[c_0,c_1]}$ denote projection onto $[c_0,c_1]$. The projected SGD update is

$$\nu_{t+1} = \Pi_{[c_0,c_1]}\big(\nu_t - \alpha' g_t\big), \qquad g_t \coloneqq 1 - z_t e^{-\nu_t}, \tag{5.102}$$

where $\{z_t\}_{t\ge 0}$ are i.i.d. copies of z and $\alpha' > 0$ is a constant step size. Note that $\mathbb{E}[g_t \mid \nu_t] = \nabla F(\nu_t) = 1 - me^{-\nu_t}$.

We present a corollary of Theorem 3.5 for SGD to minimize $F(\nu)$ below.

Corollary 5.4 *Suppose $s(\zeta) \in [c_0,c_1]$ holds and $F(\cdot)$ is L-smooth in the range of $[c_0,c_1]$. Let $\{\nu_t\}$ follow (5.102). If $\eta \le \frac{1}{L}$, Then*

$$\bar{G}_T^{\mathrm{SGD}} \coloneqq \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[F(\nu_t)-F(\nu_*)] \le \frac{(\nu_0-\nu_*)^2}{2\alpha' T} + \alpha' V'.$$

where

$$V' = \frac{\alpha'}{T}\sum_{t=0}^{T-1}(\delta_t')^2 = \frac{\mathrm{Var}(z)}{T}\sum_{t=0}^{T-1}\mathbb{E}[e^{-2\nu_t}].$$

We quantify the smoothness on the bounded domain of the objective, which introduces an exponential constant.

Lemma 5.38 *On $[c_0, c_1]$, the function $F(\nu) = me^{-\nu} + \nu$ is L-smooth with*

$$L = \sup_{\nu \in [c_0, c_1]} F''(\nu) = \sup_{\nu \in [c_0, c_1]} me^{-\nu} = me^{-c_0} = e^{\nu_* - c_0}.$$

Proof. We have $F''(\nu) = me^{-\nu}$, which is decreasing in ν, so the maximum over $[c_0, c_1]$ is attained at c_0. □

Theorem 5.17 *By choosing the optimal $\alpha' = \frac{|\nu_0 - \nu_*| e^{c_0}}{\sqrt{2T\mathrm{Var}(z)}} \le \frac{1}{L} = \frac{e^{c_0}}{m}$, SGD's upper bound becomes*

$$\bar{G}_T^{\mathrm{SGD}} \le \sqrt{2} |\nu_0 - \nu_*| \, e^{\nu_* - c_0} \sqrt{\frac{\kappa - 1}{T}}. \tag{5.103}$$

where $\kappa = \mathbb{E}[z^2]/(\mathbb{E}[z])^2$.

Proof. The proof follows Corollary 5.4 by noting that $V' \le \mathrm{Var}(z) e^{-2c_0}$ and $\mathrm{Var}(z) = m^2(\kappa - 1) = e^{2\nu_*}(\kappa - 1)$. □

Why it matters

By comparing the convergence bound of SPMD with that of SGD, the resulting ratio is:

$$\frac{1}{|\nu_0 - \nu_*| e^{\nu_* - c_0}}.$$

Notably, this ratio becomes exponentially small in regimes where $\nu_* \gg c_0$, highlighting the superior efficiency of SPMD.

Benefit under the noiseless setting

We further show that, even in the noiseless setting, the dependence of the GD update on $|\nu_0 - \nu_*|$ is unavoidable, whereas the PMD update does not exhibit such dependence when $\nu_0 \gg \nu_*$.

In the noiseless setting, where $m = \mathbb{E}[e^{s(\zeta)}]$ is known, the gradient descent (GD) iteration becomes:

$$\nu_{t+1} = \nu_t - \alpha' \nabla F(\nu_t) = \nu_t - \alpha' \left(1 - me^{-\nu_t}\right), \qquad t \ge 0, \tag{5.104}$$

where $\alpha' > 0$ is a step size. For deterministic PMD, its update is equivalent to (cf. Lemma 5.26):

$$y_{t+1} = \frac{y_t + \alpha}{1 + \alpha m}, \tag{5.105}$$

where $y_t = e^{-\nu_t}$.

Lemma 5.39 (GD vs PMD) *Assume $\nu_0 \gg \nu_*$. Let $\{\nu_t\}_{t \ge 0}$ follow (5.104) with $\alpha' \le 1$. Then in order to have $|\nabla F(\nu_t)| \le \epsilon$, then we need at least*

$$t \geq \frac{\nu_0 - \nu_* - \log\left(\frac{1}{1-\epsilon}\right)}{\alpha'}. \tag{5.106}$$

In contrast, for deterministic PMD update (5.105), in order to ensure $|\nabla F(\nu_t)| \leq \epsilon$. it suffices that

$$t = \left\lceil \frac{\log\left(|1 - r_0|/\epsilon\right)}{\log(1 + \alpha m)} \right\rceil. \tag{5.107}$$

Proof. Recall the definition $r(\nu) \coloneqq me^{-\nu} = e^{\nu_* - \nu}$. We have $|\nabla F(\nu)| = |1 - r(\nu)|$. From (5.104),

$$\nu_{t+1} = \nu_t - \alpha'\left(1 - e^{\nu_* - \nu_t}\right).$$

If $\nu_t \geq \nu_*$, then $\nu_{t+1} - \nu_* = \nu_t - \nu_* - \alpha'\left(1 - e^{\nu_* - \nu_t}\right) \geq 0$ provided $\alpha' \leq 1$. Let $r_t = e^{\nu_* - \nu_t} > 0$. Then, from GD update we have

$$r_{t+1} = r_t e^{\alpha'(1 - r_t)} \leq r_t e^{\alpha'} \leq r_0 e^{\alpha'(t+1)}.$$

In order to have $\|\nabla F(\nu_t)\|_2^2 \leq \epsilon^2$, it is necessary to have $r_t \geq 1 - \epsilon$. Hence, we need at least $t \geq \frac{\log \frac{1-\epsilon}{r_0}}{\alpha'} = \frac{\nu_0 - \nu_* - \log\left(\frac{1}{1-\varepsilon}\right)}{\alpha'}$.

For deterministic PMD update (5.105), since $r_t = my_t$ we have

$$r_{t+1} - 1 = \frac{r_t - 1}{1 + \alpha m}.$$

Taking absolute value yields

$$|\nabla F(\nu_{t+1})| = \frac{|\nabla F(\nu_t)|}{(1 + \alpha m)}.$$

Solving $|\nabla F(\nu_t)| \leq |\nabla F(\nu_0)|/(1 + \alpha m)^t \leq \epsilon$ yields (5.107). □

Why it matters

Deterministic GD needs at least $\Omega((\nu_0 - \nu_*)/\alpha')$ steps to enter a constant-accuracy region, whereas PMD reduces $|\nabla F(\nu_t)|$ geometrically with rate $(1 + \alpha m)^{-1}$, yielding a complexity of order $O\left(\frac{1}{\log(1+\alpha m)} \log \frac{1}{\epsilon}\right)$, which does not scale with ν_0 due to $|1 - r_0| = |1 - e^{\nu_* - \nu_0}| \leq 1$.
Indeed, in the noiseless setting for PMD, taking the formal limit $\alpha \to \infty$ yields $y_1 \to 1/m$ thus $\nu_1 \to \nu_*$. This highlights that the PMD update is an implicit, geometry-matched step.

5.5.2.4 An Optimal bound for SPMD

In fact, we can improve the convergence rate of SPMD to $O\left(\frac{\kappa - 1}{T}\right)$, which matches a lower bound to be established. The key is just to use a specially designed learning

rate scheme α_t. Recall the SPMD update:

$$y_t = \frac{y_{t-1} + \alpha_t}{1 + \alpha_t z_t}, \quad \forall t \geq 1, \tag{5.108}$$

where $y_{t-1} = e^{-\nu_{t-1}}, z_t = e^{s(\zeta_t)}$.

Lemma 5.40 *Let $S_t := \sum_{i=1}^t z_i$ and $\bar{z}_t := S_t/t$. Initialize $y_1 = 1/z_1$ (or equivalently $\alpha_1 = \infty$) and for $t \geq 2$ choose*

$$\alpha_t := \frac{y_{t-1}}{t-1} = \frac{1}{S_{t-1}}. \tag{5.109}$$

Then for all $t \geq 1$,

$$y_t = \frac{t}{S_t}, \qquad \nu_t = -\log y_t = \log\Big(\frac{S_t}{t}\Big) = \log \bar{z}_t. \tag{5.110}$$

In particular, ν_t is the exact minimizer of the empirical objective

$$\widehat{F}_t(\nu) := \bar{z}_t e^{-\nu} + \nu \quad \textit{since} \quad \arg\min_{\nu} \widehat{F}_t(\nu) = \log \bar{z}_t.$$

Proof. We prove (5.110) by induction. For $t = 1$, $y_1 = 1/z_1 = 1/S_1$ holds by initialization. Assume $y_{t-1} = (t-1)/S_{t-1}$. Then (5.109) gives $\alpha_t = 1/S_{t-1}$, and the recursion (5.108) yields

$$y_t = \frac{\frac{t-1}{S_{t-1}} + \frac{1}{S_{t-1}}}{1 + \frac{z_t}{S_{t-1}}} = \frac{\frac{t}{S_{t-1}}}{\frac{S_{t-1}+z_t}{S_{t-1}}} = \frac{t}{S_{t-1} + z_t} = \frac{t}{S_t}.$$

Thus $y_t = t/S_t$ and $\nu_t = -\log y_t = \log(S_t/t) = \log \bar{z}_t$. □

Assumption 5.20. *Assume $s(\zeta)$ is σ^2-subgaussian, i.e.,*

$$\mathbb{E}\big[e^{\lambda(s(\zeta) - \mathbb{E}[s(\zeta)])}\big] \leq e^{\lambda^2\sigma^2/2} \quad \forall \lambda \in \mathbb{R}.$$

This includes Bernoulli distribution (indeed, if $s(\zeta) \in [c_0, c_1]$ a.s., then $s(\zeta) - \mathbb{E}[s(\zeta)]$ is $(c_1 - c_0)^2/4$-subgaussian by Hoeffding's lemma).

Since $\frac{\mathrm{Var}(z)}{(\mathbb{E}[z])^2} = \kappa - 1$, we have

$$\mathrm{Var}(\bar{z}_T) = \frac{\mathrm{Var}(z)}{T} = \frac{(\kappa-1)m^2}{T}.$$

Since Lemma 5.40 gives $\nu_T = \log \bar{z}_T$, in light of (5.94) we can write

$$F(\nu_T) - F(\nu_*) = \frac{m}{\bar{z}_T} - 1 + \log\Big(\frac{\bar{z}_T}{m}\Big) = \frac{1}{Q_T} + \log Q_T - 1, \qquad Q_T := \frac{\bar{z}_T}{m}. \tag{5.111}$$

Note that $\mathbb{E}[Q_T] = 1$ and $\mathrm{Var}(Q_T) = (\kappa-1)/T$.

Let $U_T := Q_T - 1 = (\bar{z}_T - m)/m$. Then $\mathbb{E}[U_T] = 0$ and $\mathbb{E}[U_T^2] = (\kappa - 1)/T$. Define

$$g(u) := \frac{1}{1+u} + \log(1+u) - 1, \forall u > -1$$

so that by (5.111) we have $F(\nu_T) - F(\nu_*) = g(U_T)$.

Lemma 5.41 *For all $u \geq -\frac{1}{2}$,*

$$g(u) \leq 2u^2.$$

Proof. Define $h(u) := 2u^2 - g(u)$ for $u > -1$. Since $g'(u) = \frac{u}{(1+u)^2}$, we have

$$h'(u) = 4u - \frac{u}{(1+u)^2} = u\Big(4 - \frac{1}{(1+u)^2}\Big).$$

For $u \geq -\frac{1}{2}$, $(1+u)^2 \geq \frac{1}{4}$, hence $\frac{1}{(1+u)^2} \leq 4$. Therefore $h'(u) \leq 0$ for $u \in [-\frac{1}{2}, 0]$ and $h'(u) \geq 0$ for $u \geq 0$. Thus h attains its minimum over $[-\frac{1}{2}, \infty)$ at $u = 0$, where $h(0) = 0$. Hence $h(u) \geq 0$ on $[-\frac{1}{2}, \infty)$, i.e., $g(u) \leq 2u^2$ there. □

Lemma 5.42 *Let $z_i \geq 0$ i.i.d. with finite κ. Then*

$$\mathbb{P}(Q_T \leq 1/2) = \mathbb{P}(\bar{z}_T \leq m/2) \leq \exp\Big(-\frac{T}{8\kappa}\Big).$$

Proof. For any $\lambda > 0$, by Chernoff bound,

$$\mathbb{P}\Big(\sum_{i=1}^{T} z_i \leq \tfrac{Tm}{2}\Big) = \mathbb{P}\Big(e^{-\lambda \sum_{i=1}^{T} z_i} \geq e^{-\lambda Tm/2}\Big) \leq e^{\lambda Tm/2}\Big(\mathbb{E}[e^{-\lambda z}]\Big)^T.$$

Using $e^{-x} \leq 1 - x + x^2/2$ for $x \geq 0$,

$$\mathbb{E}[e^{-\lambda z}] \leq 1 - \lambda m + \frac{\lambda^2}{2}\mathbb{E}[z^2] \leq \exp\Big(-\lambda m + \frac{\lambda^2}{2}\mathbb{E}[z^2]\Big).$$

Therefore

$$\mathbb{P}(\bar{z}_T \leq m/2) \leq \exp\Big(T\Big(\lambda m/2 - \lambda m + \frac{\lambda^2}{2}\mathbb{E}[z^2]\Big)\Big) = \exp\Big(-T\Big(\frac{\lambda m}{2} - \frac{\lambda^2}{2}\mathbb{E}[z^2]\Big)\Big).$$

Choose $\lambda = m/(2\mathbb{E}[z^2])$ to get the exponent $-Tm^2/(8\mathbb{E}[z^2]) = -T/(8\kappa)$. □

Lemma 5.43 *If s is σ^2-subgaussian, then*

$$m^2\,\mathbb{E}[z^{-2}] \;=\; (\mathbb{E}[e^s])^2\,\mathbb{E}[e^{-2s}] \;\leq\; e^{3\sigma^2}.$$

Proof. Let $\mu = \mathbb{E}[s]$ and $X = s - \mu$. Then $\mathbb{E}[X] = 0$ and $z = e^s = e^\mu e^X$. Thus

$$m^2\mathbb{E}[z^{-2}] = \big(e^\mu \mathbb{E}[e^X]\big)^2 \cdot \big(e^{-2\mu}\mathbb{E}[e^{-2X}]\big) = \big(\mathbb{E}[e^X]\big)^2\,\mathbb{E}[e^{-2X}].$$

By subgaussianity,

$$\mathbb{E}[e^{X}] \le e^{\sigma^2/2}, \qquad \mathbb{E}[e^{-2X}] \le e^{(2^2)\sigma^2/2} = e^{2\sigma^2}.$$

Hence $m^2\mathbb{E}[z^{-2}] \le e^{\sigma^2} e^{2\sigma^2} = e^{3\sigma^2}$. □

Theorem 5.18 *Under Assumption 5.20, the SPMD iterate ν_T produced by $\alpha_t = y_{t-1}/(t-1)$ satisfies*

$$\mathbb{E}\big[F(\nu_T) - F(\nu_*)\big] \;\le\; \frac{2(\kappa-1)}{T} \;+\; e^{\frac{3}{2}\sigma^2} \exp\Big(-\frac{T}{16\kappa}\Big). \tag{5.112}$$

In particular, since the second term is exponentially small in T/κ,

$$\mathbb{E}\big[F(\nu_T) - F(\nu_*)\big] = O(\kappa/T),$$

for every σ^2-subgaussian $s(\zeta)$.

Proof. Since $F(\nu_T) - F(\nu_*) = g(U_T)$, we split the expectation on the events $\{U_T \ge -1/2\}$ and $\{U_T < -1/2\}$:

$$\mathbb{E}[g(U_T)] = \mathbb{E}[g(U_T)\mathbf{1}\{U_T \ge -1/2\}] + \mathbb{E}[g(U_T)\mathbf{1}\{U_T < -1/2\}].$$

On $\{U_T \ge -1/2\}$, Lemma 5.41 yields

$$\mathbb{E}[g(U_T)\mathbf{1}\{U_T \ge -1/2\}] \le 2\,\mathbb{E}[U_T^2] = 2\,\mathrm{Var}(Q_T) = 2\,\frac{\mathrm{Var}(z)}{m^2 T} = \frac{2(\kappa-1)}{T}.$$

On $\{U_T < -1/2\}$ we have $Q_T \le 1/2$, and since $\log Q_T - 1 \le 0$,

$$g(U_T) = \frac{1}{Q_T} + \log Q_T - 1 \le \frac{1}{Q_T}.$$

Hence, by Cauchy–Schwarz,

$$\mathbb{E}[g(U_T)\mathbf{1}\{U_T < -1/2\}] \le \mathbb{E}[Q_T^{-1}\mathbf{1}\{Q_T \le 1/2\}] \le \big(\mathbb{E}[Q_T^{-2}]\big)^{1/2}\,\mathbb{P}(Q_T \le 1/2)^{1/2}.$$

By Jensen inequality and Lemma 5.43,

$$\mathbb{E}[Q_T^{-2}] = m^2\,\mathbb{E}[\bar{z}_T^{-2}] \le m^2\,\mathbb{E}[z^{-2}] \le e^{3\sigma^2}.$$

By Lemma 5.42, $\mathbb{P}(Q_T \le 1/2) \le \exp(-T/(8\kappa))$. Therefore,

$$\mathbb{E}[g(U_T)\mathbf{1}\{U_T < -1/2\}] \le e^{\frac{3}{2}\sigma^2} \exp\Big(-\frac{T}{16\kappa}\Big).$$

Combining the two pieces proves (5.112). □

A Distribution-free lower bound

Indeed, we can show that $O\left(\frac{\kappa-1}{T}\right)$ is an optimal bound by establishing matching a lower bound for a black-box oracle model where the underlying distribution of z is unknown and for any query ν the oracle returns

$$\Phi(\nu;\zeta) = ze^{-\nu} + \nu, \qquad g(\nu;\zeta) = \nabla_\nu \Phi(\nu;\zeta) = 1 - ze^{-\nu}.$$

Since

$$z(\zeta) = e^{\nu}(\Phi(\nu;\zeta) - \nu) = e^{\nu}(1 - g(\nu;\zeta)),$$

hence, any T-query algorithm can reconstruct T i.i.d. samples $z_1, \dots, z_T$ from P. Thus, it suffices to prove the lower bound in the standard i.i.d. sampling model for z.

Let us define a distribution class. For $\kappa \geq 2$, define

$$\mathcal{P}_\kappa \coloneqq \left\{ P :\ z \geq 0,\ 0 < \mathbb{E}_P[z] < \infty,\ \frac{\mathbb{E}_P[z^2]}{(\mathbb{E}_P[z])^2} \leq \kappa \right\}.$$

Equivalently, $\mathrm{Var}_P(z)/(\mathbb{E}_P[z])^2 \leq \kappa - 1$. For $P \in \mathcal{P}_\kappa$ let $m(P) = \mathbb{E}_P[z]$ and $\nu_*(P) = \log m(P)$.

Lemma 5.44 *Let $\phi(u) \coloneqq e^{-u} + u - 1$. Then $\phi(0) = \phi'(0) = 0$ and $\phi''(u) = e^{-u}$. In particular, for all $|u| \leq 1$,*

$$\phi(u) \ \geq\ \frac{e^{-1}}{2}\, u^2. \tag{5.113}$$

Proof. On the interval $[-1, 1]$, $\phi''(u) = e^{-u} \geq e^{-1}$, so ϕ is e^{-1}-strongly convex on $[-1, 1]$. Since $\phi(0) = \phi'(0) = 0$, strong convexity implies $\phi(u) \geq \frac{e^{-1}}{2} u^2$ for all $|u| \leq 1$. □

Lemma 5.45 *Let $\phi(u) = e^{-u} + u - 1$. Fix $\nu_0 < \nu_1$ and let $\Delta \coloneqq \nu_1 - \nu_0$. Define*

$$H(\nu) \coloneqq \phi(\nu - \nu_0) + \phi(\nu - \nu_1).$$

Then H is strictly convex and its unique minimizer $\nu^\dagger$ lies in (ν_0, ν_1). Moreover, if $\Delta \leq 1$, then

$$\inf_{\nu \in \mathbb{R}} H(\nu) \ \geq\ \frac{e^{-1}}{4}\, \Delta^2. \tag{5.114}$$

Proof. We have $\phi'(u) = 1 - e^{-u}$ and $\phi''(u) = e^{-u} > 0$, hence H is strictly convex with

$$H'(\nu) = \phi'(\nu - \nu_0) + \phi'(\nu - \nu_1) = 2 - e^{-(\nu - \nu_0)} - e^{-(\nu - \nu_1)}.$$

At the endpoints,

$$H'(\nu_0) = 2 - 1 - e^{-(\nu_0 - \nu_1)} = 1 - e^{\Delta} < 0, \qquad H'(\nu_1) = 2 - e^{-(\nu_1 - \nu_0)} - 1 = 1 - e^{-\Delta} > 0.$$

Since H' is strictly increasing (because $H'' > 0$), there is a unique root $\nu^\dagger \in (\nu_0, \nu_1)$ and thus $\inf_{\nu\in\mathbb{R}} H(\nu) = \inf_{\nu\in[\nu_0,\nu_1]} H(\nu)$.

Assume $\Delta \le 1$. Then for all $\nu \in [\nu_0, \nu_1]$ we have $|\nu - \nu_0| \le \Delta \le 1$ and $|\nu - \nu_1| \le \Delta \le 1$. On $[-1, 1]$, $\phi''(u) = e^{-u} \ge e^{-1}$, so $\phi(u) \ge \frac{e^{-1}}{2}u^2$ for all $|u| \le 1$. Therefore, for all $\nu \in [\nu_0, \nu_1]$,

$$H(\nu) \ge \frac{e^{-1}}{2}\big((\nu - \nu_0)^2 + (\nu - \nu_1)^2\big).$$

Minimizing the RHS over ν yields $\inf_\nu\big((\nu - \nu_0)^2 + (\nu - \nu_1)^2\big) = \Delta^2/2$, hence $\inf_{\nu\in\mathbb{R}} H(\nu) \ge \frac{e^{-1}}{4}\Delta^2$. □

Lemma 5.46 (Le Cam's Two-point Method) *Let P_0, P_1 be two distributions and let $L_0(\cdot), L_1(\cdot)$ be nonnegative loss functions. For any estimator $\widehat{a}$ measurable w.r.t. the data,*

$$\max\{\mathbb{E}_{P_0}[L_0(\widehat{a})],\ \mathbb{E}_{P_1}[L_1(\widehat{a})]\} \ \ge\ \frac{1 - \mathrm{TV}(P_0, P_1)}{2}\ \inf_a\big(L_0(a) + L_1(a)\big). \quad (5.115)$$

Proof. Let $M \coloneqq (P_0 + P_1)/2$ and write $dP_0 = (1 + f)\,dM$, $dP_1 = (1 - f)\,dM$ where $|f| \le 1$ and $\int |f|\,dM = \mathrm{TV}(P_0, P_1)$. Then for any (possibly random) decision A,

$$\begin{aligned}
\mathbb{E}_{P_0}[L_0(A)] + \mathbb{E}_{P_1}[L_1(A)] &= \int \Big(L_0(A)(1 + f) + L_1(A)(1 - f)\Big)\,dM \\
&= \int \Big((L_0(A) + L_1(A)) + f(L_0(A) - L_1(A))\Big)\,dM \\
&\ge \int \Big((L_0(A) + L_1(A)) - |f|\,(L_0(A) + L_1(A))\Big)\,dM \\
&= \int (L_0(A) + L_1(A))(1 - |f|)\,dM \\
&\ge \inf_a(L_0(a) + L_1(a)) \int (1 - |f|)\,dM \\
&= (1 - \mathrm{TV}(P_0, P_1))\inf_a(L_0(a) + L_1(a)).
\end{aligned}$$

Taking half and using $\max\{x, y\} \ge (x + y)/2$ yields (5.115). □

The final distribution-free suboptimality lower bound is stated in the following theorem.

Theorem 5.19 *Let $z = e^{s(\zeta)} \ge 0$ with $m(P) = \mathbb{E}_P[z]$ and $\nu_*(P) = \log m(P)$. For $\kappa \ge 2$, define*

$$\mathcal{P}_\kappa \coloneqq \left\{P :\ z \ge 0,\ 0 < \mathbb{E}_P[z] < \infty,\ \frac{\mathbb{E}_P[z^2]}{\mathbb{E}_P[z]^2} \le \kappa\right\}.$$

Let $F_P(\nu) \coloneqq m(P)e^{-\nu} + \nu$ and $\nu_*(P) = \arg\min_\nu F_P(\nu)$. Then there exists an absolute constant $c > 0$ such that for all $T \ge \kappa$, any (possibly adaptive) algorithm using

T value/gradient oracle calls and outputting $\widehat{\nu}$ satisfies

$$\sup_{P\in\mathcal{P}_\kappa} \mathbb{E}_P[F_P(\widehat{\nu}) - F_P(\nu_*(P))] \;\geq\; c\,\frac{\kappa-1}{T}. \tag{5.116}$$

Proof. We construct two strictly positive hard instances in $\mathcal{P}_\kappa$. Fix $\varepsilon \in (0,1]$ and define two distributions supported on $\{\varepsilon, \kappa\}$:

$$P_i^\varepsilon: \quad \mathbb{P}(z=\kappa) = p_i, \qquad \mathbb{P}(z=\varepsilon) = 1-p_i, \qquad i\in\{0,1\},$$

where

$$p_0 \coloneqq \frac{1}{\kappa}, \qquad p_1 \coloneqq p_0 + h, \qquad h \coloneqq \frac{1}{8\sqrt{\kappa T}}.$$

Since $T \geq \kappa$, we have $h \leq \frac{1}{8\kappa}$ so $p_1 \in (0,1)$.

Next we show that $P_0^\varepsilon, P_1^\varepsilon \in \mathcal{P}_\kappa$. For a generic $p \in (0,1)$ and support $\{\varepsilon,\kappa\}$, define

$$R_\varepsilon(p) \coloneqq \frac{\mathbb{E}[z^2]}{\mathbb{E}[z]^2} = \frac{p\kappa^2 + (1-p)\varepsilon^2}{\big(p\kappa + (1-p)\varepsilon\big)^2}.$$

Let $u \coloneqq \varepsilon/\kappa \in (0, 1/\kappa] \subset (0,1]$. Then

$$R_\varepsilon(p) = \frac{p + (1-p)u^2}{\big(p + (1-p)u\big)^2}.$$

We claim $R_\varepsilon(p) \leq \frac{1}{p}$ for all $u \in [0,1]$. Indeed,

$$\begin{aligned}
&\big(p+(1-p)u\big)^2 - p\big(p+(1-p)u^2\big) \\
&= p^2 + 2p(1-p)u + (1-p)^2u^2 - p^2 - p(1-p)u^2 \\
&= (1-p)u\Big(2p + (1-2p)u\Big) \;\geq\; 0,
\end{aligned}$$

because $u \in [0,1]$ and $2p + (1-2p)u \geq \min\{2p, 1\} \geq 0$. Thus $R_\varepsilon(p) \leq 1/p$. Since $p_0 = 1/\kappa$ and $p_1 \geq p_0$, we have $1/p_i \leq \kappa$, hence $R_\varepsilon(p_i) \leq \kappa$ and therefore $P_0^\varepsilon, P_1^\varepsilon \in \mathcal{P}_\kappa$.

Next, we compute the separation Δ between ν_*'s. Let $m_i^\varepsilon = \mathbb{E}_{P_i^\varepsilon}[z] = \varepsilon + p_i(\kappa-\varepsilon)$ and $\nu_i^\varepsilon = \log m_i^\varepsilon$. Then

$$m_1^\varepsilon - m_0^\varepsilon = h(\kappa-\varepsilon) \geq h(\kappa-1), \qquad m_0^\varepsilon = \varepsilon + p_0(\kappa-\varepsilon) = 1 + \Big(1-\frac{1}{\kappa}\Big)\varepsilon \in [1,2].$$

Hence

$$\Delta \coloneqq |\nu_1^\varepsilon - \nu_0^\varepsilon| = \log\!\left(1 + \frac{m_1^\varepsilon - m_0^\varepsilon}{m_0^\varepsilon}\right) \geq \frac{1}{2}\cdot\frac{h(\kappa-1)}{2} = \frac{\kappa-1}{32\sqrt{\kappa T}},$$

where we used $\log(1+x) \geq x/2$ for $x \in [0, 1/2]$ and the fact that $\frac{h(\kappa-\varepsilon)}{m_0^\varepsilon} \leq h\kappa \leq 1/8$. In particular, $\Delta \leq h\kappa \leq 1/8 < 1$.

Next, we show the lower bound of $\inf_\nu \Big((F_0(\nu) - F_0(\nu_0^\varepsilon)) + (F_1(\nu) - F_1(\nu_1^\varepsilon)) \Big)$. Under P_i^ε the objective is $F_i(\nu) = m_i^\varepsilon e^{-\nu} + \nu$ and the optimal value is $F_i(\nu_i^\varepsilon) = 1 + \nu_i^\varepsilon$. Thus the suboptimality can be written as

$$F_i(\nu) - F_i(\nu_i^\varepsilon) = e^{\nu_i^\varepsilon - \nu} + (\nu - \nu_i^\varepsilon) - 1 = \phi(\nu - \nu_i^\varepsilon), \qquad \phi(u) = e^{-u} + u - 1.$$

Let $\nu_0^\varepsilon < \nu_1^\varepsilon$ and set $u = \nu - \nu_0^\varepsilon$. Then

$$\phi(\nu - \nu_0^\varepsilon) + \phi(\nu - \nu_1^\varepsilon) = \phi(u) + \phi(u - \Delta).$$

The function $u \mapsto \phi(u) + \phi(u - \Delta)$ is convex and its minimizer lies in $[0, \Delta]$. Since $\Delta \leq 1$, applying Lemma 5.45 gives

$$\phi(u) + \phi(u - \Delta) \;\geq\; \frac{e^{-1}}{4}\Delta^2.$$

Therefore,

$$\inf_\nu \Big((F_0(\nu) - F_0(\nu_0^\varepsilon)) + (F_1(\nu) - F_1(\nu_1^\varepsilon)) \Big) \;\geq\; \frac{e^{-1}}{4}\Delta^2. \tag{5.117}$$

Next, we show the total variation between P_0^ε, and P_1^ε is bounded. Because the two distributions differ only in the Bernoulli parameter,

$$\mathrm{KL}(P_0^\varepsilon, P_1^\varepsilon) = p_0 \log \frac{p_0}{p_1} + (1 - p_0) \log \frac{1 - p_0}{1 - p_1}.$$

Using the bound $\mathrm{KL}(P, Q) \leq \chi^2(P, Q)$ and the fact that for Bernoulli measures $\chi^2(P_0^\varepsilon, P_1^\varepsilon) = \frac{h^2}{p_1(1-p_1)}$, we get

$$\mathrm{KL}(P_0^\varepsilon, P_1^\varepsilon) \leq \frac{h^2}{p_1(1 - p_1)}.$$

Since $h \leq \frac{1}{2\kappa}$, we have $p_1 \leq p_0 + h \leq \frac{3}{2\kappa} \leq \frac{3}{4}$, hence $1 - p_1 \geq 1/4$, and also $p_1 \geq p_0 = 1/\kappa$. Therefore $p_1(1 - p_1) \geq \frac{1}{4\kappa}$ and

$$\mathrm{KL}(P_0^\varepsilon, P_1^\varepsilon) \leq 4\kappa h^2.$$

For T i.i.d. samples, this gives

$$\mathrm{KL}\big((P_0^\varepsilon)^{\otimes T}, (P_1^\varepsilon)^{\otimes T}\big) = T\,\mathrm{KL}(P_0^\varepsilon, P_1^\varepsilon) \leq 4\kappa T h^2 = \frac{1}{16}.$$

By Pinsker's inequality,

$$\mathrm{TV}\big((P_0^\varepsilon)^{\otimes T}, (P_1^\varepsilon)^{\otimes T}\big) \le \sqrt{\frac{1}{2}\mathrm{KL}\big((P_0^\varepsilon)^{\otimes T}, (P_1^\varepsilon)^{\otimes T}\big)} \le \sqrt{\frac{1}{32}} \le \frac{1}{4}.$$

Finally, we apply Lemma 5.46 to $P_0 = (P_0^\varepsilon)^{\otimes T}$, $P_1 = (P_1^\varepsilon)^{\otimes T}$ and losses

$$L_i(\nu) \coloneqq F_i(\nu) - F_i(\nu_i^\varepsilon) \ge 0.$$

Using (5.117) and $\mathrm{TV} \le 1/4$ yields for any estimator $\widehat{\nu}$,

$$\max_{i\in\{0,1\}} \mathbb{E}_{P_i^\varepsilon}\big[F_i(\widehat{\nu}) - F_i(\nu_i^\varepsilon)\big] \ge \frac{1-\mathrm{TV}}{2}\cdot\frac{e^{-1}}{4}\Delta^2 \ge \frac{3}{8}\cdot\frac{e^{-1}}{4}\Delta^2 = \frac{3e^{-1}}{32}\Delta^2.$$

Substituting $\Delta^2 \ge \frac{(\kappa-1)^2}{1024\,\kappa\,T} \ge \frac{\kappa-1}{2048\,T}$ (since $\kappa \ge 2$) gives

$$\max_{i\in\{0,1\}} \mathbb{E}_{P_i^\varepsilon}\big[F_i(\widehat{\nu}) - F_i(\nu_i^\varepsilon)\big] \;\ge\; \frac{3}{65536\,e}\cdot\frac{\kappa-1}{T}.$$

Since $P_0^\varepsilon, P_1^\varepsilon \in \mathcal{P}_\kappa$, this implies (5.116) with $c = \frac{3}{65536\,e}$. □

5.6 History and Notes

Finite-sum coupled compositional optimization (FCCO) was first formalized in our work (Qi et al., 2021c) for optimizing average precision, an empirical estimator of the area under the precision–recall curve. We proposed the SOAP algorithm for AP maximization and established the first complexity bound of $O\left(\frac{n}{\epsilon^5}\right)$ for finding an ϵ-stationary solution. Their algorithm is closely related to SOX, but differs in that it does not employ a moving-average gradient estimator. The framework was demonstrated on applications including image classification and molecular property prediction for drug discovery. The analysis of SOAP draws inspiration from the original SCGD analysis Wang et al. (2017a), while significantly improving upon its $O(1/\epsilon^8)$ complexity with the a better hyper-parameter setting, leading to Theorem 4.1.

To accelerate convergence, we subsequently adopted the moving average gradient estimator for FCCO (Wang et al., 2022). While this approach achieves a complexity order of $O\left(\frac{n}{B\epsilon^4}\right)$, it does not benefit from the variance reduction gained by using mini-batches to estimate inner function values. The limitation arises because we treat all inner functions as a single vector variable and compute a sparse unbiased stochastic estimator for this vector; consequently, the estimator does not enjoy the advantages of inner mini-batching. This improved rate and analysis was inspired by the stochastic compositional momentum method (Ghadimi et al., 2020).

Subsequently, we proposed the SOX algorithm-a significant advancement for solving FCCO (Wang and Yang, 2022), encompassing new design, theoretical analysis, and practical applications. In that work, we established a complexity of $O\left(\frac{n\sigma_0^2}{B\epsilon^4}\right)$

for SOX to find an ϵ-stationary solution in non-convex smooth FCCO problems. It integrates the analysis of stochastic block coordinate update of the $\mathbf{u}$ sequences with that of stochastic compositional momentum method.

Building on this, we developed a double-loop restarted algorithm that utilizes SOX in the inner loop to address non-convex problems under the μ-PL (Polyak-Lojasiewicz) condition, i.e., $\|\nabla F(\mathbf{w})\|_2^2 \geq \mu(F(\mathbf{w}) - \min_{\mathbf{w}} F(\mathbf{w}))$. This approach yields an improved complexity of $O\left(\frac{n\sigma_0^2}{\mu^2 B\epsilon}\right)$ for finding an ϵ-optimal solution. This result further implies a complexity of $O\left(\frac{n\sigma_0^2}{\mu^2 B\epsilon}\right)$ for strongly convex FCCO problems and $O\left(\frac{n\sigma_0^2}{B\epsilon^3}\right)$ for convex FCCO problems, requiring no assumptions on the individual convexity of inner and outer functions beyond the overall convexity of the objective. The improved convergence analysis under the PL condition for the double-loop restarted algorithm was inspired by our prior work on stochastic compositional optimization for distributionally robust learning (Qi et al., 2021b). A comparable complexity bound of $O(\frac{1}{\mu^2\epsilon})$ for a single-loop algorithm in the context of Stochastic Convex Optimization (SCO) under the PL condition was subsequently established in (Jiang et al., 2023), which considers the application of SCO in training energy-based models.

Furthermore, for convex FCCO instances where the outer function is both convex and monotonically non-decreasing and the inner functions are convex, (Wang and Yang, 2022) reformulated the problem as a convex-concave min-max optimization problem and established a complexity of $O\left(\frac{n\sigma_0^2}{B\epsilon^2}\right)$ under a weak duality convergence measure. Finally, when a μ-strongly convex regularizer is present, the complexity is further refined to $O\left(\frac{n\sigma_0^2}{\mu^2 B\epsilon}\right)$ for finding an ϵ-optimal solution in terms of Euclidean distance to the optimum. This analysis was mostly inspired by (Zhang and Lan, 2024), which is the first work that establishes the optimal complexity for solving convex SCO where the outer function is both convex and monotonically non-decreasing and the inner function is convex.

Later, Jiang et al. (2022) proposed the Multi-Block-Single-Probe Variance Reduction (MSVR) algorithm for FCCO, establishing improved complexity bounds over SOX by leveraging the mean squared smoothness of the inner functions. For non-convex smooth FCCO problems, MSVR improves the complexity to $O\left(\frac{n\sigma_0}{B\epsilon^3}\right)$ for identifying an ϵ-stationary solution.

For objectives satisfying the μ-PL condition, a double-loop restarted MSVR algorithm achieves an improved complexity of $O\left(\frac{n\sigma_0}{\mu B\epsilon}\right)$ to find an ϵ-optimal solution. Consequently, this approach yields a complexity of $O\left(\frac{n\sigma_0}{\mu B\epsilon}\right)$ for strongly convex FCCO problems and $O\left(\frac{n\sigma_0}{B\epsilon^2}\right)$ for convex FCCO problems.

The analysis for non-smooth weakly convex FCCO and the SONX (v2) algorithm was studied in our work (Hu et al., 2024b). This work established a complexity of $O\left(\frac{n\sigma_0}{B\epsilon^6}\right)$ for finding a nearly ϵ-stationary solution for weakly convex inner and outer

functions. A similar analysis for a special case of weakly-convex SCO was conducted in (Zhu et al., 2023c). When the outer function is smooth, the complexity is improved in this book to $O\left(\frac{n\sigma_0}{B\epsilon^4}\right)$. The SONEX algorithm for solving weakly convex FCCO with non-smooth outer functions was proposed in our work (Chen et al., 2025b).

The ALEXR algorithm and its analysis for convex FCCO instances appeared in our work (Wang and Yang, 2023), where the outer function is both convex and monotonically non-decreasing and the inner functions are convex. For the first time, we established a complexity of $O\left(\frac{n\sigma_0^2}{B\epsilon^2}\right)$ for finding an ϵ-optimal solution of convex FCCO. Our analysis of the stochastic block coordinate update for the dual variables is primarily informed by the framework in Alacaoglu et al. (2025), which addresses convex-concave minimax problems with bilinear structures. The extrapolation for the gradient of the dual variable is inspired by (Zhang et al., 2021). It is worth mentioning that for strongly convex FCCO with smooth outer functions, we only established the convergence of ALEXR for the Euclidean distance to the optimum. However, it is possible to establish the convergence for the objective gap and even the duality gap following our work on strongly-convex strongly-concave min-max optimization (Yan et al., 2020b).

In (Wang and Yang, 2023), we also established the lower bounds for convex FCCO and strongly convex FCCO, which matches the upper bounds. Our derivation of the lower bound for convex FCCO with non-smooth outer functions builds upon the construction presented in (Zhang and Lan, 2024) for SCO.

The double-loop ALEXR was developed in Chen et al. (2025b), which was mostly inspired by a line of work on weakly-convex concave min-max problems (Rafique et al., 2018; Yan et al., 2020b; Zhang et al., 2022). (Rafique et al., 2018) is the first work that proves the convergence for weakly-convex (strongly)-concave problems. Yan et al. (2020b) simplified the algorithm for weakly-convex strongly-concave problems with μ-strong concavity on the dual variable and established a complexity of $O(\frac{1}{\mu^2\epsilon^4})$ for finding an nearly ϵ-stationary point. The later work (Zhang et al., 2022) improved the complexity to $O(\frac{1}{\mu\epsilon^4})$ with a simple change on the number of iteration for the inner loop.

The non-convex analysis of ASGD for compositional CVaR minimization first appeared in (Zhu et al., 2022b) for one-way partial AUC optimization. The geometric-aware algorithm SCENT for CERM and its analysis were developed in (Wei et al., 2026). It remains an interesting problem to conduct fine-grained analysis of SCENT for non-convex problems.

A more general framework than FCCO is the so-called conditional stochastic optimization (CSO), defined as:

$$\min_{\mathbf{w}} \mathbb{E}_{\xi}\left[f_{\xi}\left(\mathbb{E}_{\zeta|\xi}[g(\mathbf{w};\zeta,\xi)]\right)\right].$$

This paradigm was formally introduced by Hu et al. (2020), who analyzed a biased SGD (BSGD) algorithm employing a large inner mini-batch and a constant outer mini-batch. For non-convex smooth problems, using an inner batch size of $O(\epsilon^{-2})$ results in an iteration complexity of $O(\epsilon^{-4})$, which translates to a total sample com-

plexity of $O(\epsilon^{-6})$. This performance is inferior to that of SOX when $n/B < \epsilon^{-2}$. For convex and μ-strongly convex CSO problems, an inner batch size of $O(\epsilon^{-1})$ yields iteration complexities of $O(\epsilon^{-2})$ and $O(\mu^{-2}\epsilon^{-1})$, respectively. Notably, the latter complexity is likewise worse than that of restarted SOX when $n/B < O(\epsilon^{-1})$.

Chapter 6
Applications: Learning Predictive, Generative and Representation Models

Abstract In this chapter, we present applications of stochastic compositional optimization and finite-sum coupled compositional optimization (FCCO) in both supervised and self-supervised learning settings. These include training predictive models, generative models, and representation models based on advanced objective functions such as distributionally robust optimization (DRO), group DRO (GDRO), AUC losses, NDCG loss, and contrastive losses. We also highlight applications of compositional optimization in solving multiple inequality-constrained optimization problems, optimizing data compositional neural networks, and a new paradigm of learning with a reference model called DRRHO risk minimization.

Unity of knowledge and action!

Contents

Algorithm 23 Stochastic Optimization Framework of DL

// The Meta Algorithm
1: Set the learning rate schedule η_t
2: **for** $t = 1, \cdots, T$ **do**
3: Compute a vanilla gradient estimator $\mathbf{z}_t$
4: Update $\mathbf{w}_{t+1}$ by calling the update of SGD, Momentum, Adam, or AdamW optimizer
5: **end for**

// The SGD optimizer update
1: Update $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{z}_t$

// The Momentum optimizer update
1: Update $\mathbf{v}_t = \beta_1 \mathbf{v}_{t-1} + (1-\beta_1)\mathbf{z}_t$ ⋄ the MA gradient estimator
2: Update $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{v}_t$

// The Adam optimizer update
1: Update $\mathbf{v}_t = \beta_1 \mathbf{v}_{t-1} + (1-\beta_1)\mathbf{z}_t$ ⋄ the MA gradient estimator
2: Update $\mathbf{s}_t = \beta_2 \mathbf{s}_{t-1} + (1-\beta_2)(\mathbf{z}_t)^2$
3: Update $\hat{\mathbf{v}}_t = \mathbf{v}_t / (1-\beta_1^t)$
4: Update $\hat{\mathbf{s}}_t = \mathbf{s}_t / (1-\beta_2^t)$
5: Update $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \frac{\hat{\mathbf{v}}_t}{\sqrt{\hat{\mathbf{s}}_t}+\epsilon}$ ϵ is a small constant

// The AdamW optimizer update
1: Update $\mathbf{v}_t = \beta_1 \mathbf{v}_{t-1} + (1-\beta_1)\mathbf{z}_t$ ⋄ the MA gradient estimator
2: Update $\mathbf{s}_t = \beta_2 \mathbf{s}_{t-1} + (1-\beta_2)(\mathbf{z}_t)^2$
3: Update $\hat{\mathbf{v}}_t = \mathbf{v}_t / (1-\beta_1^t)$
4: Update $\hat{\mathbf{s}}_t = \mathbf{s}_t / (1-\beta_2^t)$
5: Update $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \left(\frac{\hat{\mathbf{v}}_t}{\sqrt{\hat{\mathbf{s}}_t}+\epsilon} + \lambda \mathbf{w}_t \right)$ λ is a weight-decay constant

6.1 Stochastic Optimization Framework

For practioners who may skip Chapter 3, Chapter 4, and Chapter 5, we first provide a brief introduction to the stochastic optimization framework commonly used for deep learning. We also highlight the challenges in solving advanced machine learning problems introduced in Chapter 2 and summarize the key ideas behind the solution methods presented in Chapters 4 and 5.

The standard procedure for implementing a stochastic optimization algorithm typically involves computing a vanilla gradient estimator, followed by updating the model parameters using a step of an optimizer. We present a meta-algorithm in Algorithm 23, along with four classical optimizers: SGD, Momentum, Adam, and AdamW.

Three forms of the Momentum Method

The Momentum method represents a key milestone (as further discussed in the next subsection). The stochastic momentum method originates from the Heavy-ball (HB) method, whose stochastic version (SHB) has the following update for solving $\min_{\mathbf{w}} F(\mathbf{w}) := \mathbb{E}_{\zeta}[f(\mathbf{w};\zeta)]$:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \nabla f(\mathbf{w}_t;\zeta_t) + \beta_1(\mathbf{w}_t - \mathbf{w}_{t-1}), \tag{6.1}$$

where $\beta_1 \in (0,1)$ is the momentum parameter. While we utilize a single stochastic gradient $\nabla f(\mathbf{w}_t;\zeta_t)$ for illustrative purposes, practical applications generally rely on mini-batch estimation. In Section 4.3, we show it is equivalent to the the following update with moving average gradient estimator:

$$\begin{aligned} \mathbf{v}_t &= \beta_1 \mathbf{v}_{t-1} + (1-\beta_1)\nabla f(\mathbf{w}_t;\zeta_t) \\ \mathbf{w}_{t+1} &= \mathbf{w}_t - \eta' \mathbf{v}_t, \end{aligned} \tag{6.2}$$

Update (6.1) is equivalent to (6.2) if $\eta'(1-\beta_1) = \eta$. In PyTorch, the Momentum method is implemented by the following update:

$$\begin{aligned} \mathbf{v}_t &= \beta_1 \mathbf{v}_{t-1} + \nabla f(\mathbf{w}_t;\zeta_t) \\ \mathbf{w}_{t+1} &= \mathbf{w}_t - \eta \mathbf{v}_t, \end{aligned} \tag{6.3}$$

which is equivalent to (6.1). One key insight from the convergence analysis of the Momentum method (6.2) (cf. Theorem 4.3) is that it ensures the averaged estimation error of the moving-average gradient estimators $\{\mathbf{v}_t\}$ converge to zero.

Thanks to well-developed deep learning frameworks such as PyTorch, implementing training code for deep neural networks has become relatively straightforward. The standard training pipeline is shown in Figure 6.1. The `Dataset` module allows us to get a training sample, which includes its input and output. The `Data Sampler` module (typically wrapped within the `DataLoader` module) provides tools to sample a mini-batch of examples for training at each iteration. The `Model` module allows us to define different deep models. The `Mini-batch Loss` module defines a loss function on the selected mini-batch data for backpropagation. The `Optimizer` module implements methods for updating the model parameter given the computed gradient from backpropagation. Most essential functions are already available in PyTorch. In practice, users often only need to define a function to compute their mini-batch losses. By calling `loss.backward()`, a mini-batch stochastic gradient, serving as a vanilla gradient estimator, is computed automatically.

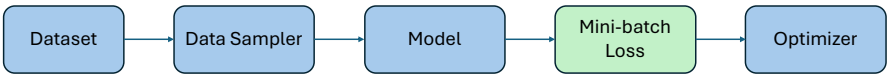


Fig. 6.1: Standard training pipeline for deep learning. Users typically only need to implement the mini-batch loss function. It relies on a critical assumption that the mini-batch stochastic gradient is an unbiased estimator of the true gradient

6.1.1 Milestones of Stochastic Optimization

While the Adam optimizer has become a standard in machine learning as of 2025, it has deep roots in the innovations of stochastic optimization before deep learning era. Below, we briefly discuss key milestones of stochastic optimization that have impact on the Adam method.

Stochasticity. The fundamental concept of gradient descent (GD), dating back to (Cauchy, 1847), uses the full dataset's gradient to take a step in the steepest direction. Introduced by Robbins and Monro (1951), SGD improves upon GD by using only a small batch of data (or even a single data point) to estimate the gradient, significantly speeding up training on large datasets.

Acceleration. To improve the convergence rate of GD, Polyak (1964) proposed the Heavy-ball (HB) method, which itself originates from the second-order Richardson method for solving a system of linear equations (Frankel, 1950). While Polyak only proved a faster rate of local convergence than GD for smooth and strongly convex problems, Nemirovski and Yudin (1977) proved the first nearly optimal rate for general smooth and strongly convex problems. Their method was inspired by the conjugate gradient method for solving quadratic problems and needs to solve 2-dimensional optimization problem using the method of centers of gravity every step; cf. (Nemirovsky and Yudin, 1983)[Sec. 7.3]. Later, Nesterov (1983) derived a simpler form of accelerated gradient method, which is now known as Nesterov's accelerated gradient (NAG) method.

Nesterov's Accelerated Gradient (NAG) method

The original update form of the NAG method is given by:

$$\begin{aligned} \mathbf{u}_{t+1} &= \mathbf{w}_t - \eta \nabla F(\mathbf{w}_t), \\ \mathbf{w}_{t+1} &= \mathbf{u}_{t+1} + \beta_1(\mathbf{u}_{t+1} - \mathbf{u}_t). \end{aligned} \tag{6.4}$$

It is equivalent to

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \nabla F(\mathbf{w}_t) + \beta_1((\mathbf{w}_t - \eta \nabla F(\mathbf{w}_t)) - (\mathbf{w}_{t-1} - \eta \nabla F(\mathbf{w}_{t-1}))). \tag{6.5}$$

Comparing with the HB method (6.1), the momentum term is changed from $\beta(\mathbf{w}_t - \mathbf{w}_{t-1})$ to $\beta(\mathbf{u}_{t+1} - \mathbf{u}_t)$.
If we let $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \mathbf{v}_t$, then the NAG update is equivalent to

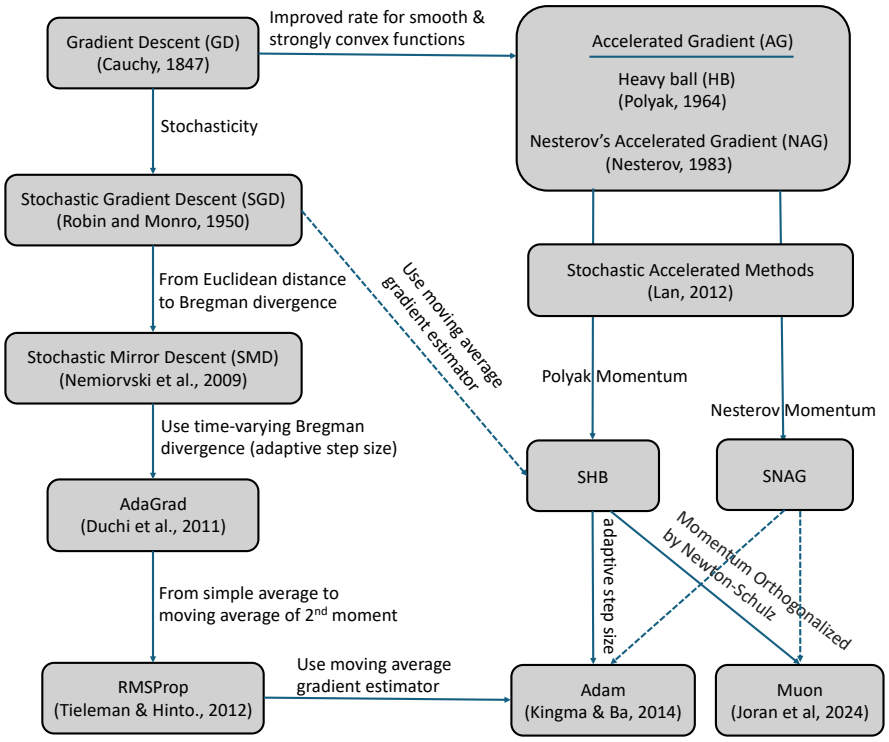


Fig. 6.2: Evolution of Stochastic Optimization

$$\begin{aligned}\mathbf{v}_t &= \beta_1 \mathbf{v}_{t-1} + \nabla F(\mathbf{w}_t) + \beta_1(\nabla F(\mathbf{w}_t) - \nabla F(\mathbf{w}_{t-1})) \\ \mathbf{w}_{t+1} &= \mathbf{w}_t - \eta \mathbf{v}_t .\end{aligned} \tag{6.6}$$

This is similar to (6.3) except that an error correction term $\beta(\nabla F(\mathbf{w}_t) - \nabla F(\mathbf{w}_{t-1}))$ is added to the gradient estimator update.
We can also make the updates in (6.4) or (6.6) stochastic, leading to the stochastic NAG (SNAG) method. In particular, if we use a stochastic gradient estimator $\nabla f(\mathbf{w}_t; \zeta_t)$ in (6.4), we have the following update:

$$\begin{aligned}\mathbf{u}_{t+1} &= \mathbf{w}_t - \eta \nabla f(\mathbf{w}_t; \zeta_t), \\ \mathbf{w}_{t+1} &= \mathbf{u}_{t+1} + \beta_1(\mathbf{u}_{t+1} - \mathbf{u}_t).\end{aligned} \tag{6.7}$$

If we use stochastic gradient estimators $\nabla f(\mathbf{w}_t; \zeta_t)$ and $\nabla f(\mathbf{w}_{t-1}; \zeta_t)$ in (6.6), we have the following update:

$$\begin{aligned} \mathbf{v}_t &= \beta_1 \mathbf{v}_{t-1} + \nabla f(\mathbf{w}_t; \zeta_t) + \beta_1 (\nabla f(\mathbf{w}_t; \zeta_t) - \nabla f(\mathbf{w}_{t-1}; \zeta_t)) \\ \mathbf{w}_{t+1} &= \mathbf{w}_t - \eta \mathbf{v}_t. \end{aligned} \tag{6.8}$$

The difference between the two variants lies that (6.8) needs to compute two stochastic gradient estimators at $\mathbf{w}_t$ and $\mathbf{w}_{t-1}$ per-iteration. However, interested readers can show that the update in (6.8) with a variable change is equivalent to the STORM update as presented in Section 4.3.2 for optimizing $F(\mathbf{w}) = \mathbb{E}_\zeta [f(\mathbf{w}; \zeta)]$.

Lan (2012) pioneered the development and analysis of stochastic accelerated gradient methods, achieving the optimal rates in both deterministic and stochastic regimes. Its update is slightly different from the NAG update. (Yang et al., 2016) is the first work to prove the convergence of stochastic NAG and stochastic HB methods for non-convex optimization.

Adaptive step sizes. The technique of utilizing coordinate-wise adaptive step sizes was pioneered by AdaGrad (Duchi et al., 2011), a method whose analysis is rooted in the framework of Stochastic Mirror Descent (SMD) (Nemirovski et al., 2009). Both AdaGrad and SMD are thoroughly examined in Chapter 3. RMSProp, appeared in a course lecture (Tieleman and Hinton, 2012), moved from AdaGrad's simple average of the second moment (squared gradients) to a moving average of the second moment. The moving average estimator has a long history in stochastic optimization, see (Ermoliev and Wets, 1988)[Sec. 6.2.3]. Finally, RMSProp leads to the current standard, the Adam method (Kingma and Ba, 2014), which combines the moving average of the first moment (similar to SHB) with the moving average of the second moment (similar to RMSProp). AdamW is a variant of Adam, which decouples weight decay from gradient-based updates.

Recently, a new optimizer named Muon (Jordan et al., 2024) has emerged, specifically designed to optimize matrix-structured parameters, such as the weight matrices between neural network layers. In contrast, conventional optimizers typically treat these parameters as flattened vectors, potentially overlooking their inherent structural properties.

The Muon method

Let W_t denote a matrix-structured parameter at the t-th iteration. The Muon update is given by:

$$\begin{aligned} M_t &= \beta_1 M_{t-1} + \nabla f(W_t; \zeta_t) \\ (U_t, S_t, V_t) &= \mathrm{SVD}(M_t) \\ W_{t+1} &= W_t - \eta_t U_t V_t^\top. \end{aligned} \tag{6.9}$$

In practice, the Singular Value Decomposition (SVD) is often replaced by a more computationally efficient Newton-Schulz matrix iteration. This process produces an approximate matrix $O_t = U_t S_t' V_t^\top$, where S_t' is diagonal with

$S'_t[i,i]' \sim \text{Uniform}(0.5, 1.5)$. The weight update is then applied as $W_{t+1} = W_t - \eta_t O_t$.

Summary: The evolution of stochastic optimization, which has had a major impact on modern AI (see Figure 6.2), can be characterized by five key shifts in algorithm design:

- From Full Gradient to Stochastic Gradient (**Batch Size**): Switched from using the full dataset's gradient (GD) to using noisy stochastic gradients (SGD) for faster iteration speed.
- From Gradient Descent to Accelerated Gradient Methods (**Momentum**): The optimization technique was enhanced by introducing a momentum term (like HB or NAG) to achieve an improved convergence rate for smooth convex functions, while still using the full gradient.
- From Euclidean Distance to Bregman Divergence (**Geometry**): Switched the underlying distance metric used for updates from the Euclidean distance to a Bregman divergence (SMD).
- From Static Step Size to Adaptive Step Size (**Preconditioning**): Switched from a constant or manually decaying learning rate to one that is scaled by past gradient magnitudes (AdaGrad).
- From a Mini-batch gradient estimator to a Moving Average gradient estimator (**Error reduction**): Switched from a simple mini-batch gradient estimator to a moving average gradient estimator (SHB, Adam).

6.1.2 Limitations of Existing Optimization Framework

The standard stochastic optimization algorithms and their analyses rest on a critical assumption: that the mini-batch stochastic gradient is an unbiased estimator of the true gradient. As discussed in Chapter 4, this assumption breaks down in the case of compositional functions of the form $f(g(\mathbf{w}))$, where f is a deterministic non-linear function and g is a stochastic function. In such cases, the gradient of the mini-batch loss $f(g(\mathbf{w}; \mathcal{B}))$, where $g(\mathbf{w}; \mathcal{B})$ is an unbiased estimator of $g(\mathbf{w})$ with a mini-batch $\mathcal{B}$, yields a biased estimate of the true gradient. Specifically, calling `loss.backward()` on the mini-batch loss will return a gradient of $\nabla f(g(\mathbf{w}; \mathcal{B}))\nabla g(\mathbf{w}; \mathcal{B})$, which is inherently biased. The method that directly uses this biased gradient estimator for SGD update is referred to as biased SGD (BSGD). However, since the estimation error is inversely proportional to the batch size, small batches can lead to large optimization errors. According to Lemma 2.1, such errors can negatively impact the generalization performance of the learned model.

To address this challenge, Chapters 4 and 5 introduce solution methods tailored to different families of compositional objectives. The key ideas underlying these algorithms concern (i) how the vanilla gradient estimator $\mathbf{z}_t$ is computed in Step 3 of Algorithm 23, and (ii) how the estimator error is further reduced through the use

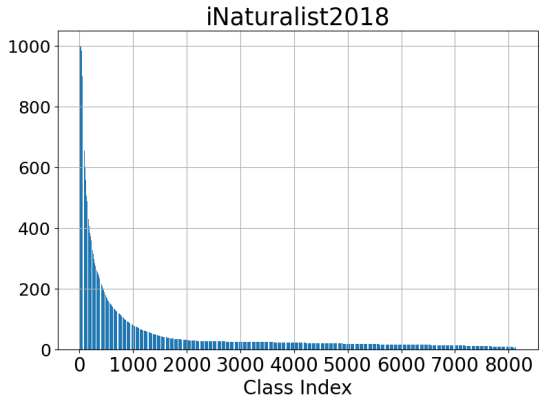


Fig. 6.3: Histograms of class sizes of the iNaturalist2018 dataset, which contains 437,513 natural images of 8,142 species. The sizes of classes follow a long-tail distribution.

of moving-average (MA) estimators $\mathbf{v}_t$ as in Step 1 of the Momentum optimizer or more advanced variance-reduction techniques. In the following sections, we will present their applications to various complex and advanced machine learning problems, with a focus on the presentation of the novel vanilla gradient estimators, which allow us to integrate them into the standard optimization schemes such as Momentum or AdamW for non-convex deep learning problems.

6.2 DRO and Group DRO

Let us consider supervised learning with a set of training data $\{(\mathbf{x}, y)\}$, where $\mathbf{x} \in \mathbb{R}^d$ denotes the input data and $y \in \{1, \ldots, K\}$ denotes the output class label. Let $\ell(\mathbf{w}; \mathbf{x}, y)$ denote the pointwise loss function, e.g., the cross-entropy loss.

6.2.1 DRO for Imbalanced Classification

Imbalanced classification is prevalent in many areas, including medicine and cybersecurity, where most training data may belong to one or a few classes. Mathematically, it means that the marginal distribution of the class label is a non-uniform distribution. An example of an imbalanced dataset is shown in Figure 6.3.

For imbalanced data, the conventional empirical risk minimization would focus on minimizing the loss of data from those dominating classes, neglecting data from the minority classes. DRO can address this issue by assigning larger weights to data

with higher losses. Let us first consider the KL-divergence regularized DRO:

$$\min_{\mathbf{w}} \max_{\mathbf{p}\in\Delta_n} \sum_{i=1}^n p_i \ell(\mathbf{w}; \mathbf{x}_i, y_i) - \tau \sum_{i=1}^n p_i \log(p_i n) + r(\mathbf{w}), \tag{6.10}$$

where $r(\mathbf{w})$ is a regularizer on $\mathbf{w}$. A traditional way to solve this problem is to use stochastic minimax optimization algorithms. However, there are several drawbacks of this approach: (1) the variance of stochastic gradient for $\mathbf{w}$ depends on the sampling distribution and the best sampling distribution depends on $\mathbf{p}$; (2) the sampling of data based on $\mathbf{p}$ incurs additional costs and is not friendly to practical implementation that uses random shuffling; (3) stochastic update of the dual variable $\mathbf{p}$ either takes $O(n)$ time complexity per iteration or requires maintaining a special tree structure to reduce the updating time to $O(\log(n))$.

To circumvent these issues, we consider an alternative formulation that is equivalent to the above minimax objective, i.e.,

$$\min_{\mathbf{w}} \tau \log\left(\frac{1}{n}\sum_{i=1}^n \exp\left(\frac{\ell(\mathbf{w}; \mathbf{x}_i, y_i)}{\tau}\right)\right) + r(\mathbf{w}). \tag{6.11}$$

For simplicity, we just consider the standard Euclidean norm regularization $r(\mathbf{w}) = \frac{\lambda}{2}\|\mathbf{w}\|_2^2$. As a result, the first term in the objective takes the form of a compositional optimization problem, namely $f\big(\mathbb{E}_\zeta\left[\, g(\mathbf{w}; \zeta)\,\right]\big)$, where $f(\cdot) = \tau\log(\cdot)$ and

$$\mathbb{E}_\zeta\left[(g(\mathbf{w}; \zeta)\right] = \frac{1}{n}\sum_{i=1}^n \exp\left(\frac{\ell(\mathbf{w}; \mathbf{x}_i, y_i)}{\tau}\right).$$

The SCGD, SCMA, SCST, and SCENT algorithms can be applied to solve the above problem. We now focus on the application of SCMA, whose key steps are presented in Algorithm 24.

The vanilla gradient estimator $\mathbf{z}_t$ of the first term in (6.11) at the t-th iteration is computed by :

$$\mathbf{z}_t = \frac{1}{B}\sum_{i\in\mathcal{B}_t} \frac{\exp(\frac{\ell(\mathbf{w}_t; \mathbf{x}_i, y_i)}{\tau})}{u_t} \nabla\ell(\mathbf{w}_t; \mathbf{x}_i, y_i). \tag{6.12}$$

It is motivated from (4.4) where the same mini-batch $\mathcal{B}_t$ is used for both updating u_t and computing $\mathbf{z}_t$.

Let us compare this gradient estimator with that of stochastic optimization for empirical risk minimization:

$$\hat{\mathbf{z}}_t = \frac{1}{B}\sum_{i\in\mathcal{B}_t} \nabla\ell(\mathbf{w}_t; \mathbf{x}_i, y_i). \tag{6.13}$$

The difference between (6.12) and (6.13) lies in the blue term, which acts as a weight for each data in the mini-batch. In the vanilla gradient estimator $\mathbf{z}_t$ for DRO, the data

Algorithm 24 Attentional Biased Stochastic Methods

1: **for** $t = 1, \cdots, T$ **do**
2: Sample a mini-batch of B samples $\mathcal{B}_t \subset [n]$
3: Compute $g(\mathbf{w}_t, \mathcal{B}_t) = \frac{1}{B}\sum_{i\in\mathcal{B}_t} \exp(\ell(\mathbf{w}_t; \mathbf{x}_i, y_i)/\tau)$
4: Compute $u_t = (1-\gamma)u_{t-1} + \gamma g(\mathbf{w}_t, \mathcal{B}_t)$
5: Compute the vanilla gradient estimator $\mathbf{z}_t = \frac{1}{B}\sum\limits_{i\in\mathcal{B}_t} \frac{\exp(\frac{\ell(\mathbf{w}_t;\mathbf{x}_i,y_i)}{\tau})}{u_t}\nabla\ell(\mathbf{w}_t; \mathbf{x}_i, y_i)$
6: Update $\mathbf{w}_{t+1}$ by an optimizer such as Momentum or Adam-W
7: **end for**

in the mini-batch with a larger loss $\ell(\mathbf{w}_t; \mathbf{x}_i, y_i)$ has a higher weight. This will facilitate the learning for data from the minority group. Due to this effect, we also refer to Algorithm 24 as attentional biased stochastic method, named as AB-xx depending on which optimizer is used.

The use of u_t for normalization to compute the weight $\exp(\ell(\mathbf{w}_t; \mathbf{x}_i, y_i)/\tau)/u_t$ is also different from that using the heuristic mini-batch normalization where the weight is computed by $\frac{\exp(\ell(\mathbf{w}_t;\mathbf{x}_i,y_i)/\tau)}{\sum_{i\in\mathcal{B}_t}\exp(\ell(\mathbf{w}_t;\mathbf{x}_i,y_i)/\tau)}$, which does not ensure convergence if the batch size is not significantly large. Let us consider a simple case such that only one data is sampled for updating. In this case, the mini-batch normalization gives a weight 1 for the selected data no matter whether it is from the majority or minority class. However, if the sampled data denoted by $(\mathbf{x}_t, y_t)$ at the t-th iteration is from a minority group and hence has a large loss, we would like to penalize more on such an example. The estimator $u_t = (1-\gamma)u_{t-1} + \gamma\exp(\ell(\mathbf{w}_t; \mathbf{x}_t, y_t)/\tau)$ is likely to be smaller than $\exp(\ell(\mathbf{w}_t; \mathbf{x}_t, y_t)/\tau)$ as $\gamma < 1$. As a result, normalization using u_t will give a larger weight to the sampled minority data compared with using the mini-batch normalization, i.e., $\exp(\ell(\mathbf{w}_t; \mathbf{z}_t)/\tau)/u_t > 1$. Qi et al. (2020) empirically demonstrated that using $\gamma < 1$ outperforms the case $\gamma = 1$, which corresponds to using the standard mini-batch loss.

To illustrate the effect of AB-momentum on imbalanced data. We present an experiment on synthetic data in Figure 6.4, which compares the result of using the Momentum method for ERM and AB-momentum for solving KL-divergence regularized DRO. Figure 6.4(d) shows that AB-momentum learns a better decision boundary than that of the Momentum method for ERM. Figure 6.4(b) shows that data from the minority group that are close to the decision boundary get higher weights during the training.

Practical Tips

We discuss several practical tips for computing $\mathbf{z}_t$ and other variants of DRO in the context of deep learning.

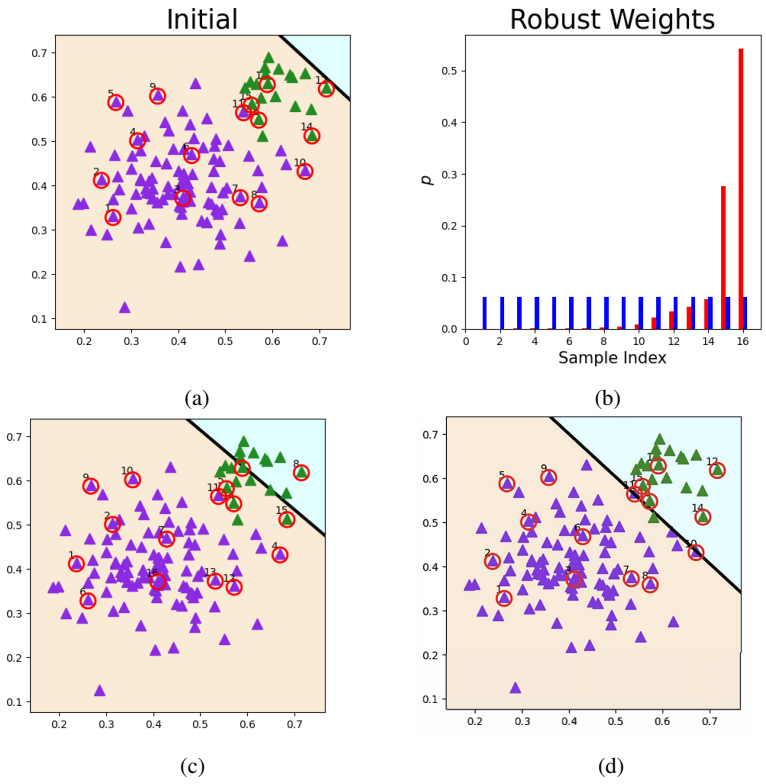


Fig. 6.4: **(a)**: A synthetic data for imbalanced binary classification (green vs purple) with a random linear decision boundary (black line). **(c), (d)**: Learned linear models optimized by the standard momentum method for ERM and AB-momentum for DRO with logistic loss for 100 iterations, respectively. **(b)**: The averaged weights of circled samples in the training process of the standard momentum method for ERM and AB-momentum method for DRO. Sample with indices in $\{1, \ldots, 11\}$ are from the majority class and samples with indices in $\{12, 13, 14, 15, 16\}$ are from the minority class with sample 15, 16 close to the decision boundary.

Backpropagation.

In order to compute the vanilla gradient estimator $\mathbf{z}_t$ using the PyTorch backward function, we just need to have a slight change of computing the loss based on the mini-batch data. Below we give the pseudo code in PyTorch for computing the gradient estimator highlighted in Step 5 of Algorithm 24. It is worth noting that the line of `p=(exp_loss/u).detach()` calculates the blue part and detaches it from the computational graph so that gradient is not computed again for it. With the gradient estimator computed by `loss.backward()`, then we can use any existing optimizers, including the Momentum method and AdamW.

```
sur_loss=surrogate_loss(preds, labels)
exp_loss = torch.exp(sur_loss/tau)
u = (1 - gamma)*u + gamma*(exp_loss.mean())
p = (exp_loss/u).detach()
loss = torch.mean(p * sur_loss)
loss.backward()
```

Avoiding the numerical issue.

However, a numerical issue may arise during the running tied to the computation of $\exp(\ell(\mathbf{w}_t; \mathbf{x}_i, y_i)/\tau)$, especially when τ is small and the loss function of selected data is large so that overflow. As a result, the running of the algorithm may crash due to a NaN error. To address this issue, we maintain $\nu_t = \log u_t$. Specifically, we denote by $q_{t,i} = \exp(\frac{\ell(\mathbf{w}_t; \mathbf{x}_i, y_i) - \ell_{\max,t}}{\tau})$, where $\ell_{\max,t} = \max_{i \in \mathcal{B}_t} \ell(\mathbf{w}_t; \mathbf{x}_i, y_i)$. Then Step 4 can be reformulated to:

$$\begin{aligned}\exp(\log u_t) =& \exp(\log(1-\gamma) + \log u_{t-1}) \\ &+ \exp\left(\log\gamma + \log\left(\frac{1}{B}\sum\nolimits_{i\in\mathcal{B}_t} q_{t,i}\right) + \frac{\ell_{\max,t}}{\tau}\right).\end{aligned}$$

For simplicity, let $b_t = \log(1-\gamma) + \log u_{t-1}$ and $q_t = \log\gamma + \log\left(\frac{1}{B}\sum_{i\in\mathcal{B}_t} q_{t,i}\right) + \frac{\ell_{\max,t}}{\tau}$, we have

$$\exp(\log u_t) = \exp(b_t) + \exp(q_t).$$

The update in equivalent to following:

$$\begin{aligned}&\exp(\log u_t) = \exp(\max\{b_t, q_t\})(1 + \exp(-|b_t - q_t|)) \\ &= \exp(\max\{b_t, q_t\})\sigma^{-1}(|b_t - q_t|),\end{aligned}$$

where $\sigma(\cdot)$ denotes the sigmoid function. Taking the log on both sides gives the update for $\log u_t$. To summarize, we maintain and update $\nu_t = \log u_t$ as following:

$$\begin{aligned}b_t &= \log(1-\gamma) + \nu_{t-1} \\ q_t &= \log\gamma + \log\left(\frac{1}{B}\sum\nolimits_{i\in\mathcal{B}_t}\exp\left(\frac{\ell(\mathbf{w}_t; \mathbf{x}_i, y_i) - \ell_{\max,t}}{\tau}\right)\right) + \frac{\ell_{\max,t}}{\tau} \\ \nu_t &= \max\{b_t, q_t\} - \log\sigma(|b_t - q_t|).\end{aligned} \tag{6.14}$$

At the first iteration $t = 1$, we can just set

$$\nu_1 = \log\left(\frac{1}{B}\sum\nolimits_{i\in\mathcal{B}_t}\exp\left(\frac{\ell(\mathbf{w}_1; \mathbf{x}_i, y_i)}{\tau} - \frac{\ell_{\max,1}}{\tau}\right)\right) + \frac{\ell_{\max,1}}{\tau}.$$

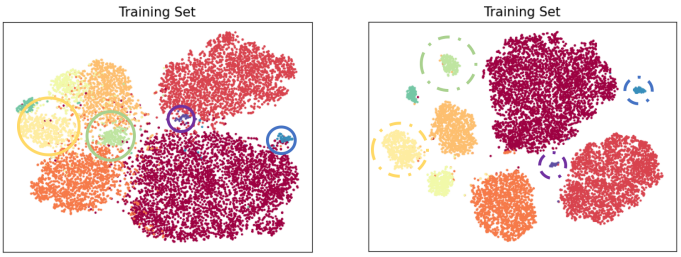


Fig. 6.5: t-SNE visualization of feature representations of training & testing set on CIFAR10-LT ($\rho = 100$) with different strategies of setting τ. Right: Fixed $\tau = 1$. Left: Two-stage decay of τ: first phase $\tau = 100$ and second phase $\tau = 1$. For more details, please refer to (Qi et al., 2020).

With ν_t, the effective weight $\frac{\exp(\ell(\mathbf{w}_t;\mathbf{x}_i,y_i)/\tau)}{u_t}$ can be computed by

$$\frac{\exp\left(\frac{\ell(\mathbf{w}_t;\mathbf{x}_i,y_i)}{\tau} - \max(\frac{\ell_{\max,t}}{\tau}, \nu_t)\right)}{\exp\left(\nu_t - \max(\frac{\ell_{\max,t}}{\tau}, \nu_t)\right)}.$$

Thus, all computation involving $\exp(\cdot)$ will not incure any numerical issue.

The Temperature parameter.

The last point we discuss here is how to set the value of the temperature parameter τ. A simple way is to treat it as a hyper-parameter and tune it based on cross-validation. However, there is a trade-off in the performance. A deep neural network is a hierarchical learner with lower layers for low-level feature extraction, middle layers for more abstract feature extraction and the last layer for classification. A larger τ indicates a more uniform weight, which is not good for learning the last classifier layer and minority class specific features. A smaller τ indicates a more non-uniform weight, which is not good for learning class agnostic lower level features.

One approach to mitigate this issue is to use a two-stage approach. In the first stage, we can use a relatively larger temperature τ for learning class agnostic lower level features. The second stage, we decrease τ to finetune the upper layers for learning robust minority-class specific features and classifier layer. An example is shown in Figure 6.5 on a long-tailed version of the CIFAR10 dataset, where the data is intentionally made imbalanced such that the number of samples per class follows a long-tail distribution, the imbalance ratio ρ means the ratio between sample sizes of the most frequent and least frequent classes.

Another approach is to treat τ as a parameter to be optimized. To achieve this, we can consider optimizing a KL-divergence constrained DRO:

$$\begin{aligned} &\min_{\mathbf{w}} \max_{\mathbf{p}\in\Delta_n} \sum_{i=1}^n p_i \ell(\mathbf{w}; \mathbf{x}_i, y_i) - \tau_0 \sum_{i=1}^n p_i \log(p_i n) + r(\mathbf{w}), \\ &\text{s.t.} \quad \sum_{i=1}^n p_i \log(p_i n) \le \rho, \end{aligned} \tag{6.15}$$

where the regularizer term with a small τ_0 is added to avoid ill conditioning, making the resulting problem smooth in terms of losses. Using the dual form of the maximization problem (see (2.19)), the above problem is equivalent to

$$\min_{\mathbf{w}, \tau \ge \tau_0} \tau \log\left(\frac{1}{n}\sum_{i=1}^n \exp\left(\frac{\ell(\mathbf{w}; \mathbf{x}_i, y_i)}{\tau}\right)\right) + \tau\rho. \tag{6.16}$$

We can extend Algorithm 24 to optimize the above problem by treating $(\mathbf{w}, \tau)$ as a single variable to be optimized. The vanilla gradient estimator in terms of τ at the t-th iteration is given by :

$$\mathbf{z}_{\tau,t} = \log(u_t) + \rho - \frac{1}{B}\sum_{i\in\mathcal{B}_t} \frac{\exp(\frac{\ell(\mathbf{w}_t;\mathbf{x}_i,y_i)}{\tau_t})}{u_t} \frac{\ell(\mathbf{w}_t; \mathbf{x}_i, y_i)}{\tau_t}.$$

6.2.2 GDRO for Addressing Spurious Correlation

Data may exhibit imbalance not in the marginal distribution of class label but some joint distribution of the class label and some attributes. Please see a discussion on the example of classifying waterbird images from landbirds images in Section 2.2.3. As a consequence, the model may learn spurious correlations between the labels and some attributes. GDRO can be used to mitigate this issue by leveraging prior knowledge of spurious correlations to define groups over the training data.

Formally, if there is spurious correlation between class label $y \in \mathcal{Y}$ and some attribute $a \in \mathcal{A}$, we can group the training data into $|\mathcal{Y}| \times |\mathcal{A}|$ groups according to the value of (y, a). Let $\mathcal{D}_i = \{(\mathbf{x}_{i,j}, y_{i,j})\}_{j=1}^{n_i}$ denote the data from the i-th group for $i \in \{1, \dots K\}$. Then we can define the averaged loss for data from each group i as $L_i(\mathbf{w}) = \frac{1}{n_i}\sum_{j=1}^{n_i} \ell(\mathbf{w}; \mathbf{x}_{i,j}, y_{i,j})$. Then, the GDRO formulation with CVaR divergence corresponding to the top-k groups is equivalent to (cf. (2.26)):

$$\min_{\mathbf{w},\nu} \frac{1}{K}\sum_{i=1}^K [L_i(\mathbf{w}) - \nu]_+ + \alpha\nu + \frac{\lambda}{2}\|\mathbf{w}\|_2^2, \tag{6.17}$$

where $\alpha = \frac{k}{K}$. If we define $\bar{\mathbf{w}} = (\mathbf{w}, \nu)$ and the inner functions as $g(\bar{\mathbf{w}}) = L_j(\mathbf{w}) - \nu$ and the outer function as $f(g) = [g]_+$, then the problem becomes an instance of non-smooth FCCO, where the outer function is non-smooth.

Algorithm 25 SONEX for solving (6.18)

Input: learning rate schedules $\{\eta_t\}_{t=1}^T$, $\{\gamma_t\}_{t=1}^T$; starting points $\mathbf{w}_1$, $\mathbf{u}_0$
for $t = 1, \ldots, T$ **do**
  Draw a batch of B_1 groups $\mathcal{B}_t \subset [K]$
  for $i \in \mathcal{B}_t$ **do**
    Draw B_2 samples $\zeta_{i,t}^j \sim \mathcal{D}_i$, $j = 1, \ldots, B_2$
    Update the inner function value estimators by

$$u_{i,t} = (1 - \gamma_t) u_{i,t-1} + \gamma_t \frac{1}{B_2} \sum_{j=1}^{B_2} \ell(\mathbf{w}_t; \mathbf{x}_{i,j}, y_{i,j})$$

  end for
  Set $u_{i,t+1} = u_{i,t}$, $i \notin \mathcal{B}_t$
  Compute the vanilla gradient of ν_t: $\mathbf{z}_{t,\nu} = -\frac{1}{B_1} \sum_{i \in \mathcal{B}_t} \nabla f_\varepsilon(u_{i,t} - \nu_t) + \frac{k}{K}$
  Compute the vanilla gradient of $\mathbf{w}_t$:

$$\mathbf{z}_{t,\mathbf{w}} = \frac{1}{B_1} \sum_{i \in \mathcal{B}_t} \left(\nabla f_\varepsilon(u_{i,t} - \nu_t) \frac{1}{B_2} \sum_{j=1}^{B_2} \nabla \ell(\mathbf{w}_t; \mathbf{x}_{i,j}, y_{i,j}) \right)$$

  update ν_{t+1} using SGD
  Update $\mathbf{w}_{t+1}$ using Momentum or AdamW
end for

An alternative way is to formulate the problem into an equivalent min-max formulation:

$$\min_{\mathbf{w}} \max_{\mathbf{p} \in \Delta, n p_i \leq 1/\alpha} \sum_{i=1}^K p_i L_i(\mathbf{w}) + \frac{\lambda}{2} \|\mathbf{w}\|_2^2. \tag{6.18}$$

However, solving this min-max problem has similar drawbacks as discussed in DRO, especially when the number of groups K is large.

Let us discuss the applicability of algorithms presented in Chapter 4 for solving (6.17). The theory of SOX and MSVR requires the smoothness of the outer functions, which is not applicable to GDRO. Both ALEXR and SONX are applicable as their analysis does not require the smoothness of the outer functions. However, their updates is SGD-type, which could make it slow or fail in practice for learning modern deep neural networks such as Transformer.

For deep learning applications, we can leverage SONEX. Its key idea is to smooth the outer hinge function. In particular, we define the smoothed hinge function as $f_\varepsilon(g)$ with a very small ε (cf. Example 5.1):

$$f_\varepsilon(g) = \max_{y \in [0,1]} yg - \frac{\varepsilon}{2} y^2 = \begin{cases} g - \frac{\varepsilon}{2} & \text{if } g \geq \varepsilon \\ \frac{g^2}{2\varepsilon} & \text{if } 0 < g < \varepsilon \\ 0 & \text{o.w.} \end{cases}.$$

As a result, we solve the following smoothed problem:

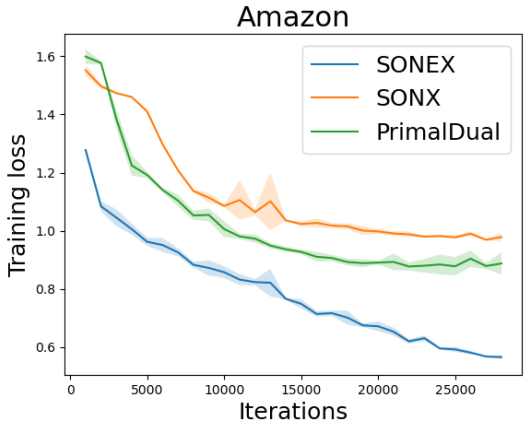


Fig. 6.6: An experimental comparison of different methods for solving GDRO (2.26) on the Amazon-WILDS dataset. The dataset is a text classification benchmark derived from Amazon product reviews, where the task is to predict binary sentiment (positive or negative) using TF–IDF features extracted from review text. The data spans multiple product categories. We construct groups based on the user attribute, resulting in 1,252 distinct groups. Only 4 groups and 64 data points per-group are sampled per-iteration. SONEX uses the Adam optimizer, SONX uses the SGD optimizer, and the PrimalDual is a stochastic primal-dual method for solving (6.18) that uses the Adam optimizer for the primal variable (model weights) and uses the stochastic mirror descent update for the dual variable $\mathbf{p}$ with a KL divergence. For more details, please refer to (Chen et al., 2025b).

$$\min_{\mathbf{w},\nu} \frac{1}{K}\sum_{j=1}^{K} f_\varepsilon(L_j(\mathbf{w}) - \nu) + \alpha\nu + \frac{\lambda}{2}\|\mathbf{w}\|_2^2. \tag{6.19}$$

We present a variant of SONEX in Algorithm 25. Figure 6.6 illustrates the effectiveness of SONEX for solving GDRO comprising with SONX and a stochastic primal-dual method.

6.3 Extreme Multi-class Classification

Multi-class classification is a cornerstone of machine learning. However, many modern applications involve an exceptionally large label space—ranging from millions to even billions of categories—a challenge known as extreme multi-class classification (XMC). For instance, for face recognition, the model learning is often formulated as classifying images into unique identities. With millions of distinct individuals, the model must navigate millions of corresponding classes. Similarly, when training a language model to predict the next word, the problem is treated as a multi-class classification task where each word in the vocabulary represents a category. Given that the English language contains over one million words, the resulting number of classes is immense.

Algorithm 26 The SCENT Algorithm for solving XMC

1: Initialize W_1, ν_0, step sizes η_t and α_t, $\varphi(\nu) = e^{-\nu}$.
2: **for** $t = 1 \ldots, T-1$ **do**
3: Sample a mini-batch data $\mathcal{B}_t \subset \{1, \ldots, n\}$ with $|\mathcal{B}_t| = B$
4: Let C_t denote the set of unique labels in $\mathcal{B}_t$
5: **for** each $(\mathbf{x}_i, y_i) \in \mathcal{B}_t$ **do**
6: Update $\nu_{i,t}$ by solving

$$\nu_{i,t} = \arg\min_{\nu} \frac{1}{|\mathcal{B}_t| - 1} \sum_{y_j \in \mathcal{B}_t \setminus y_i} \exp((\mathbf{w}_{t,y_j} - \mathbf{w}_{t,y_i})^\top h(\mathbf{x}_i) - \nu) + \nu + \frac{1}{\alpha_t} D_\varphi(\nu, \nu_{i,t-1})$$

7: **end for**
8: Compute $\mathbf{Z}_t[C_t] = \nabla L_t(W_t[C_t])$ by calling backprop on the mini-batch loss

$$L_t(W_t[C_t]) = \frac{1}{B} \sum_{i \in \mathcal{B}_t} \frac{1}{|\mathcal{B}_t| - 1} \sum_{y_j \in \mathcal{B}_t \setminus y_i} \exp((\mathbf{w}_{t,y_j} - \mathbf{w}_{t,y_i})^\top h(\mathbf{x}_i) - \nu_{i,t})$$

9: Compute $\mathbf{V}_t[C_t] = (1 - \beta_t)\mathbf{V}_{t-1}[C_t] + \beta_t \mathbf{Z}_t[C_t]$ (optional)
10: Update $W_{t+1}[C_t] = W_t[C_t] - \eta_t \mathbf{V}_t[C_t]$
11: **end for**

A dominating approach of multi-class classification is logistic regression, which minimizes the cross-entropy loss. Let us consider learning a linear model by solving the following problem:

$$\min_W \frac{1}{n} \sum_{i=1}^{n} -\log \frac{\exp(\mathbf{w}_{y_i}^\top h(\mathbf{x}_i))}{\sum_{j=1}^{K} \exp(\mathbf{w}_j^\top h(\mathbf{x}_i))}$$

where $y_i \in \{1, \ldots, K\}$ denotes the true class label of $\mathbf{x}_i$, $W = (\mathbf{w}_1, \ldots, \mathbf{w}_K) \in \mathbb{R}^{d \times K}$ contains the weights for all classes, and $h(\mathbf{x}) \in \mathbb{R}^d$ denotes the feature vector of each data. When K is huge, it is not efficient to compute the normalization term $\sum_{j=1}^{K} \exp(\mathbf{w}_j^\top h(\mathbf{x}_i))$ for each data and loading all W into the memory might be prohibited.

To solve this problem, we can use SCENT algorithm presented in Section 5.5.2. To this end, we reformulate the problem into the following equivalent min-min optimization:

$$\min_W \min_\nu \frac{1}{n} \sum_{i=1}^{n} \left\{ \frac{1}{K} \sum_{j=1}^{K} \exp(\mathbf{w}_j^\top h(\mathbf{x}_i) - \mathbf{w}_{y_i}^\top h(\mathbf{x}_i) - \nu_i) + \nu_i - 1 \right\}.$$

We present an application of SCENT for solving this problem in Algorithm 26. At each iteration, the algorithm begins by sampling a mini-batch $\mathcal{B}_t$ (Step 3) to approximate the outer summation over n data points. Following this, the algorithm updates the dual variables ν_i for each $i \in \mathcal{B}_t$. While the original SCENT algorithm requires sampling from the full set of classes $\{j = 1, \ldots, K\}$, we observe that for all sampled data, the weights corresponding to their true labels $\{\mathbf{w}_{y_i} : i \in \mathcal{B}_t\}$ must already

be accessed. Consequently, we utilize the 'in-batch' class labels to approximate the inner summation, setting $\mathcal{Y}_t = \{\{y_i\}\}_{i\in\mathcal{B}_t}$ be the multiset of labels and C_t to the set of unique labels in $\mathcal{B}_t$. To update $\boldsymbol{\nu}_t$ and W_t, the following calculations are implemented.

- **Computing Sampled and Shifted Logits.** Given the mini-batch $\mathcal{B}_t$ and the set of sampled classes $\mathcal{Y}_t$, we first compute the inner products between the features $h(\mathbf{x}_i)$ and class weights $\mathbf{w}_j$ for all $i \in \mathcal{B}_t$ and $j \in \mathcal{Y}_t$. This is efficiently computed via the matrix product $Q = H[\mathcal{B}_t]^\top W[\mathcal{Y}_t] \in \mathbb{R}^{B\times|\mathcal{Y}_t|}$, where $H[\mathcal{B}_t] = [h(\mathbf{x}_i)]_{i\in\mathcal{B}_t}$ represents the sampled feature matrix. We then derive the shifted logits matrix R, defined by the entries $R_{ij} = \mathbf{w}_j^\top h(\mathbf{x}_i) - \mathbf{w}_{y_i}^\top h(\mathbf{x}_i)$ for all $i \in \mathcal{B}_t, j \in \mathcal{Y}_t$.
- **Closed-form update for $\nu_{i,t}$.** Given the shifted logits matrix R, we update the state variable $\nu_{i,t}$ according to Lemma 5.26:

$$\nu_{i,t} = \nu_{i,t-1} + \log\left(1 + \alpha_t \frac{1}{|\mathcal{Y}_t| - 1} \sum_{j\in\mathcal{Y}_t\setminus y_i} \exp(R_{ij})\right) - \log(1 + \alpha_t e^{\nu_{i,t-1}}),$$

  where we treat the labels in $\mathcal{Y}_t \setminus y_i$ as independent samples from $\{1, \ldots, K\}$. To ensure numerical stability when $\nu_{i,t-1}$ or R_{ij} are large, we apply standard logarithmic identities. Specifically, while $\nu_{i,t-1}$ typically remains within a stable range, the term $\log(1+\alpha_t e^{\nu_{i,t-1}})$ can be computed as $\nu_{i,t-1}+\log(e^{-\nu_{i,t-1}}+\alpha_t)$ for large positive values of $\nu_{i,t-1}$. Furthermore, we stabilize the second term using the Log-Sum-Exp trick by shifting the exponents by $R_{i,\max} = \max_{j\in\mathcal{Y}_t\setminus y_i} R_{ij}$:

$$\begin{aligned}
&\log\left(1 + \frac{\alpha_t}{|\mathcal{Y}_t| - 1} \sum_{j\in\mathcal{Y}_t\setminus y_i} \exp(R_{ij})\right) \\
&= \log\left(\exp(-R_{i,\max}) + \frac{\alpha_t}{|\mathcal{Y}_t| - 1} \sum_{j\in\mathcal{Y}_t\setminus y_i} \exp(R_{ij} - R_{i,\max})\right) + R_{i,\max}.
\end{aligned}$$

- **Updating $W_t[C_t]$.** Finally, the gradient of $W_t[C_t]$ is computed by performing backpropagation on the mini-batch loss $L_t(W_t[C_t])$. Because the loss function is defined only over the sampled classes, the gradient updates are sparse and operate exclusively on the sampled subset $W_t[C_t]$. This approach eliminates the need to load the entire weight matrix W into the main memory, significantly reducing the memory overhead in hardware-constrained environments.

Empirical Comparison with baselines

An empirical study demonstrating the effectiveness of SCENT for XMC is presented in Figure 6.7, which compares Algorithm 26 with ASGD, BSGD, and the SOX method. The key differences between these methods and Algorithm 26 are as follows: (i) SOX is closely related to SCENT, but uses a step size $\alpha_{i,t} = \gamma e^{-\nu_{i,t-1}}$ when

updating $\nu_{i,t}$; (ii) ASGD employs a standard stochastic coordinate update for the dual variables $\boldsymbol{\nu}$; and (iii) BSGD simply computes the gradient of $W_t[C_t]$ using the following mini-batch loss:

$$\frac{1}{B}\sum_{i\in\mathcal{B}_t} -\log\frac{\exp(\mathbf{w}_{y_i}^\top h(\mathbf{x}_i))}{\sum_{j\in\mathcal{Y}_t\setminus y_i}\exp(\mathbf{w}_j^\top h(\mathbf{x}_i))}.$$

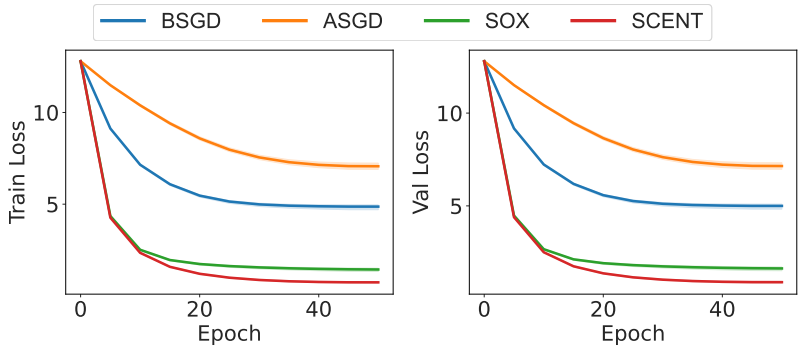


Fig. 6.7: Left: training curve on Glint360K dataset. Right: accuracy curve on the validation data. The Glint360K dataset (An et al., 2021) is a face recognition dataset consisting of 17 million images of 360 thousand individuals (i.e., 360K classes). To obtain the features for linear classification, we leverage a pretrained ResNet-50 model. For all the methods, we use a batch size of 1024 and update the model weights for 50 epochs using the SGD optimizer (no momentum). We tune the learning rate of W for all methods and decrease it in a cosine manner during training. For ASGD, SOX and SCENT, the learning rate of the $\boldsymbol{\nu}$ update is also tuned. For more details, please refer to (Wei et al., 2026).

6.4 Stochastic AUC and NDCG Maximization

In many domains such as radiology and drug discovery, areas under the curves are commonly used to assess the performance of a predictive model. In domains that involve ranking or recommendation, normalized discounted cumulative gain (NDCG) is commonly used as a performance metric. We present applications of SCO and FCCO algorithms for optimizing these metrics directly.

6.4.1 Stochastic AUC Maximization

In this section, we focus on optimizing the area under ROC curve (AUC) for binary classification as depicted in Figure 2.3.

Method 1: Pairwise Loss Minimization

The training data consists of $\{\mathbf{x}_i, y_i\}_{i=1}^n$, where $\mathbf{x} \in \mathbb{R}^d$ is the input and $y \in \{1, -1\}$ is the binary label. The traditional surrogate objective for AUC maximization is the pairwise loss given in (2.31). To optimize the pairwise surrogate objective, we just need to sample positive and negative data and then define a mini-batch pairwise loss:

$$\frac{1}{|\mathcal{B}_+|}\sum_{\mathbf{x}_i \in \mathcal{B}_+} \frac{1}{|\mathcal{B}_-|}\sum_{\mathbf{x}_j \in \mathcal{B}_-} \ell(h(\mathbf{w}; \mathbf{x}_j) - h(\mathbf{w}; \mathbf{x}_i)).$$

Calling backpropagation on this mini-batch pairwise loss gives an unbiased stochastic gradient estimator. Then any appropriate optimizer can be leveraged to update the model. This is same as the conventional algorithm except for that the data sampler needs to sample both positive and negative data (see Section 6.4.5).

A limitation of this approach is that it increases the communication costs of distributed training when data are distributed across different machines as it requires to form positive-negative pairs across different machines.

Method 2: Minimax Optimization

The second approach is to solve the formulation as in (2.32). To illustrate the algorithm, we give its formulation below:

$$\begin{aligned}\min_{\mathbf{w}\in\mathbb{R}^d,(a,b)\in\mathbb{R}^2} \quad & \frac{1}{|\mathcal{S}_+|}\sum_{\mathbf{x}_i \in \mathcal{S}_+} (h(\mathbf{w}; \mathbf{x}_i) - a)^2 + \frac{1}{|\mathcal{S}_-|}\sum_{\mathbf{x}_j \in \mathcal{S}_-} (h(\mathbf{w}; \mathbf{x}_j) - b)^2 \\ & + f\left(\frac{1}{|\mathcal{S}_-|}\sum_{\mathbf{x}_j \in \mathcal{S}_-} h(\mathbf{w}; \mathbf{x}_j) - \frac{1}{|\mathcal{S}_+|}\sum_{\mathbf{x}_i \in \mathcal{S}_+} h(\mathbf{w}; \mathbf{x}_i)\right),\end{aligned} \tag{6.20}$$

where $h(\mathbf{w}; \cdot) \in \mathbb{R}$ is the prediction output of the model for any input, $\mathcal{S}_+$ is the set of positive data and $\mathcal{S}_-$ is the set of negative data and f is a non-decreasing surrogate function.

Let us illustrate the algorithm for a squared-hinge surrogate function $f(s) = \max(m + s, 0)^2$, where $m > 0$ is a margin parameter. Since f is non-linear, the last term of the above objective function is a compositional function of the form $f(g)$, where $g(\mathbf{w}) = \frac{1}{|\mathcal{S}_-|}\sum_{\mathbf{x}_j \in \mathcal{S}_-} h(\mathbf{w}; \mathbf{x}_j) - \frac{1}{|\mathcal{S}_+|}\sum_{\mathbf{x}_i \in \mathcal{S}_+} h(\mathbf{w}; \mathbf{x}_i)$. We consider the minimax reformulation similar to (5.27). In particular, using the conjugate of $f(\cdot)$

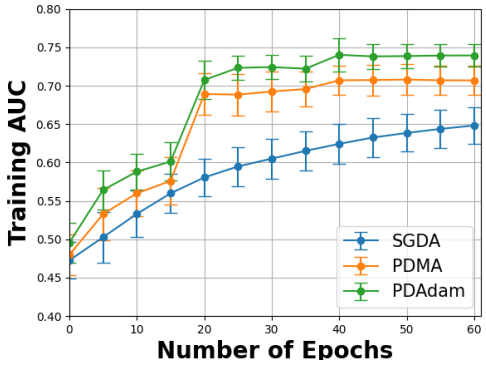


Fig. 6.8: Comparison between PDMA/PDAdam and SGDA for solving (6.21) of AUC maximization. The dataset is BBBP whose task is to predict whether a drug can penetrate the blood-brain barrier to arrive the targeted central nervous system or not. For more details, please refer to (Guo et al., 2021b).

(see Example 1.12), we convert the above minimization problem into a minimax optimization problem:

$$\min_{\mathbf{w},a,b}\max_{\alpha\geq 0} F(\mathbf{w},a,b;\alpha) := \frac{1}{|\mathcal{S}_+|}\sum_{\mathbf{x}_i\in\mathcal{S}_+}(h(\mathbf{w};\mathbf{x}_i)-a)^2 + \frac{1}{|\mathcal{S}_-|}\sum_{\mathbf{x}_j\in\mathcal{S}_-}(h(\mathbf{w};\mathbf{x}_j)-b)^2 + \alpha\left(m+\frac{1}{|\mathcal{S}_-|}\sum_{\mathbf{x}_j\in\mathcal{S}_-}h(\mathbf{w};\mathbf{x}_j)-\frac{1}{|\mathcal{S}_+|}\sum_{\mathbf{x}_i\in\mathcal{S}_+}h(\mathbf{w};\mathbf{x}_i)\right)-\frac{\alpha^2}{4}, \tag{6.21}$$

Compared to pairwise loss minimization, the advantage of the above minimax formulation is that its objective is decomposable over individual data points, making it well-suited for distributed training.

We present a practical framework in Algorithm 27 built from SMDA for solving the above problem, where the primal-dual Momentum method (PDMA) employs the momentum update for the primal variable $\bar{\mathbf{w}}$ or a primal-dual Adam method (PDAdam) employs the Adam update for the primal variable. The effectiveness of PDMA/PDAdam over SGDA for solving (6.21) on a real-world dataset is shown in Figure 6.8.

Squared-hinge surrogate vs Square surrogate function

The minimax optimization framework (6.20) and PDMA/PDAdam algorithms with a small modification on the dual variable update can handle any smooth surrogate function f. When $f(s) = (m+s)^2$ is a square surrogate, the minimax formulation is equivalent to the pairwise loss minimization with a square surrogate loss (AUC square loss). Nevertheless, the minimax AUC margin loss with the squared-hinge surrogate is more robust than the AUC square loss. Figure 6.9 illustrates the robustness of the minimax AUC margin loss.

Algorithm 27 PDMA or PDAdam for solving (6.21)

1: **Input:** learning rate schedules η_t, τ_t; starting points $\bar{\mathbf{w}}_1 = (\mathbf{w}_1, a_1, b_1)$, α_1
2: **for** $t = 1, \ldots, T$ **do**
3: Draw B_1 positive data $\mathcal{B}_t^+ \subset \mathcal{S}_+$ and B_2 negative data $\mathcal{B}_t^- \subset \mathcal{S}_-$
4: Update $\alpha_{t+1} = \left[(1 - \tau_t/2)\alpha_t + \tau_t \left(m + \frac{1}{B_2}\sum_{\mathbf{x}_j \in \mathcal{B}_t^-} h(\mathbf{w}; \mathbf{x}_j) - \frac{1}{B_1}\sum_{\mathbf{x}_i \in \mathcal{B}_t^+} h(\mathbf{w}_t; \mathbf{x}_i)\right)\right]_+$
5: Compute the vanilla gradient estimator

$$\mathbf{z}_t = \frac{1}{B_1}\sum_{i \in \mathcal{B}_t^+} \nabla_{\bar{\mathbf{w}}}(h_{\mathbf{w}_t}(\mathbf{x}_i) - a_t)^2 + \frac{1}{B_2}\sum_{\mathbf{x}_j \in \mathcal{B}_t^-} \nabla_{\bar{\mathbf{w}}}(h(\mathbf{w}_t; \mathbf{x}_j) - b_t)^2$$
$$+ \alpha_t \nabla_{\bar{\mathbf{w}}}\left(\frac{1}{B_2}\sum_{\mathbf{x}_j \in \mathcal{B}_t^-} h(\mathbf{w}; \mathbf{x}_j) - \frac{1}{B_1}\sum_{\mathbf{x}_i \in \mathcal{B}_t^+} h(\mathbf{w}_t; \mathbf{x}_i)\right)$$

6: Update $\bar{\mathbf{w}}_{t+1}$ by Momentum or AdamW
7: **end for**

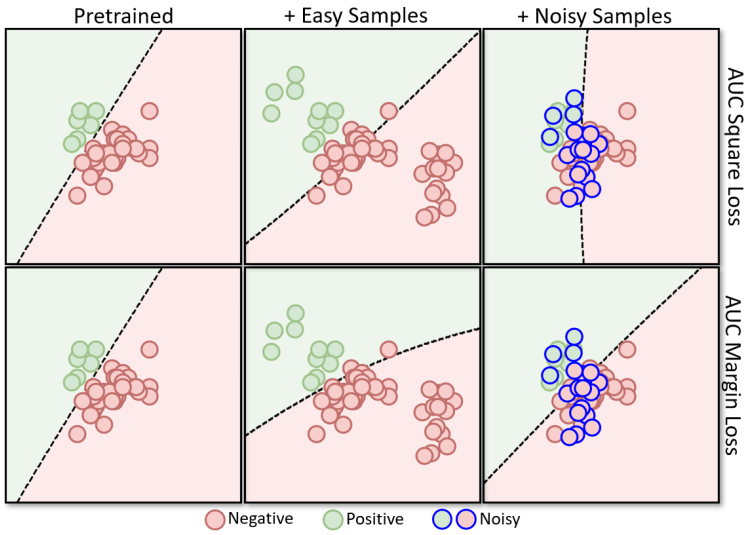


Fig. 6.9: An illustrative example for optimizing different AUC losses on a toy data for learning a two-layer neural network with ELU activation. The top row is optimizing the AUC square loss and the bottom row is optimizing the new AUC margin loss as in (6.21). The first column depicts the initial decision boundary (dashed line) pre-trained on a set of examples. In the middle column, we add some easy examples to the training set and retrain the model by optimizing the AUC loss. In the last column, we add some noisily labeled data (blue circled data) to the training set and retrain the model by optimizing the AUC loss. The results demonstrate the AUC margin loss is more robust than the AUC square loss.

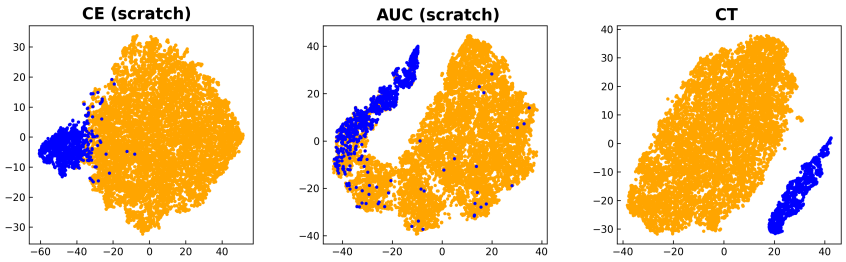


Fig. 6.10: t-SNE visualization of feature representations of an imbalanced training set for the Cat vs Dog visualized by t-SNE learned by different methods (from left to right): optimizing CE loss, an AUC loss, and a compositional training (CT) objective. For more details, please refer to (Yuan et al., 2022a).

Feature Learning

Feature learning is an important capability of deep learning. However, like the DRO objective, the end-to-end training based on the AUC surrogate objective does not favor feature learning as compared with traditional ERM. The reason is that AUC surrogate objective gives unequal weights to different data points due to the imbalance of training data. To address this challenge, one way is to employ a two-stage approach, where the first stage pretrains the encoder network on the training data by traditional supervised learning (e.g., ERM with the CE loss) or self-supervised representation learning and the second stage fine-tunes the feature extration layers and a random initialized classifier layer by optimizing an AUC surrogate objective.

An approach for performing effective feature learning and AUC maximization in a unified framework is to optimize a compositional objective (Yuan et al., 2022a):

$$\min_{\mathbf{w},a,b} \max_{\alpha \geq 0} F(\mathbf{w} - \tau \nabla L_{\text{CE}}(\mathbf{w}), a, b; \alpha),$$

where $L_{\text{CE}}(\mathbf{w})$ is the empirical risk based on the CE loss and $\tau > 0$ is a hyper-parameter.

To understand this compositional objective intuitively, let us take a thought experiment by using a gradient descent method to optimize the compositional objective. To this end, we denote the objective by $L_{\text{AUC}}(\mathbf{w} - \tau \nabla L_{\text{CE}}(\mathbf{w}))$, where L_{AUC} denotes the AUC surrogate objective. First, we evaluate the inner function by $\mathbf{u} = \mathbf{w} - \alpha \nabla L_{\text{CE}}(\mathbf{w})$. We can see that $\mathbf{u}$ is computed by a gradient descent step for minimizing the empirical risk $L_{\text{CE}}(\mathbf{w})$, which facilitates the learning of lower layers for feature extraction due to equal weights of all examples. Then, we take a gradient descent step to update $\mathbf{w}$ for minimizing the outer function $L_{\text{AUC}}(\cdot)$ by using the gradient $\nabla L_{\text{AUC}}(\mathbf{u})$ instead of $\nabla L_{\text{AUC}}(\mathbf{w})$. Because $\mathbf{u}$ is better than $\mathbf{w}$ in terms of feature extraction layers, taking a gradient descent step using $\nabla L_{\text{AUC}}(\mathbf{u})$ would be better than using $\nabla L_{\text{AUC}}(\mathbf{w})$. In addition, taking a gradient descent step for the outer function $L_{\text{AUC}}(\cdot)$ will make the classifier more robust to the minority class due to use of the AUC surrogate loss. Overall, we have two alternating conceptual steps, i.e., the inner gradient descent

step $\mathbf{u} = \mathbf{w} - \tau\nabla L_{\text{CE}}(\mathbf{w})$ acts as a feature purification step, and the outer gradient descent step $\mathbf{w} - \eta(I - \tau\nabla^2 L_{\text{CE}}(\mathbf{w}))\nabla L_{\text{AUC}}(\mathbf{u})$ acts as a classifier robustification step, where η is a step size.

For practical implementation, the intermediate model $\mathbf{w} - \tau\nabla L_{\text{CE}}(\mathbf{w})$ can be tracked by the MA estimator $\mathbf{u}_t = (1-\gamma)\mathbf{u}_{t-1} + \gamma(\mathbf{w}_t - \tau\nabla\hat{L}_{\text{CE}}(\mathbf{w}_t))$, where $\hat{L}_{\text{CE}}$ is a mini-batch CE loss. Then, $\mathbf{u}_t$ is used to update the primal variables $(\mathbf{w}; a; b)$ and the dual variable α.

Finally, we remark that the data sampler is different from traditional one because it needs to sample both positive and negative examples. It also has great impact on the performance. We defer the discussion to section 6.4.5.

6.4.2 Stochastic AP Maximization

Using a surrogate loss, AP maximization can be formulated as an FCCO problem (2.36), i.e.,

$$\min_{\mathbf{w}} \frac{1}{n} \sum_{\mathbf{x}_i \in \mathcal{S}_+} f(\mathbf{g}(\mathbf{w}; \mathbf{x}_i, \mathcal{S})), \tag{6.22}$$

where $\mathcal{S}_+$ denotes the set of n positive examples, $\mathcal{S}$ is the set of all examples, and

$$\begin{aligned}
&f(\mathbf{g}) = -\frac{[\mathbf{g}]_1}{[\mathbf{g}]_2}, \\
&\mathbf{g}(\mathbf{w}; \mathbf{x}_i, \mathcal{S}) = [g_1(\mathbf{w}; \mathbf{x}_i, \mathcal{S}), g_2(\mathbf{w}; \mathbf{x}_i, \mathcal{S})], \\
&g_1(\mathbf{w}; \mathbf{x}_i, \mathcal{S}) = \frac{1}{|\mathcal{S}|} \sum_{\mathbf{x}_j \in \mathcal{S}} \mathbb{I}(y_j = 1)\ell(h(\mathbf{w}; \mathbf{x}_j) - h(\mathbf{w}; \mathbf{x}_i)), \\
&g_2(\mathbf{w}; \mathbf{x}_i, \mathcal{S}) = \frac{1}{|\mathcal{S}|} \sum_{\mathbf{x}_j \in \mathcal{S}} \ell(h(\mathbf{w}; \mathbf{x}_j) - h(\mathbf{w}; \mathbf{x}_i)),
\end{aligned}$$

where $\ell(\cdot)$ is a non-decreasing surrogate pairwise loss (see examples in Table 2.3).

We present an application of SOX to solving the above problem in Algorithm 28, which is referred to as SOAP.

Initialization of u

Unlike traditional algorithms, Algorithm 28 for AP maximization requires initializing an additional set of auxiliary variables $\mathbf{u}_1, \ldots, \mathbf{u}_n$. In contrast to the model parameter $\mathbf{w}$, which is randomly initialized, these auxiliary variables can be initialized upon their first update. Specifically, when index i is first sampled, we set $\mathbf{u}_{i,t-1}$ to the corresponding mini-batch estimator of the inner function value. As a result,

Algorithm 28 The SOAP algorithm for AP maximization (6.22)

Input: learning rate schedules $\{\eta_t\}_{t=1}^T$, $\{\gamma_t\}_{t=1}^T$; starting points $\mathbf{w}_1$, $\mathbf{u}_0$
for $t = 1, \ldots, T$ **do**
  Draw B_1 positive data $\mathcal{B}_t^+ \subset \mathcal{S}_+$ and B_2 negative data $\mathcal{B}_t^- \subset \mathcal{S}_-$
  for $\mathbf{x}_i \in \mathcal{B}_t^+$ **do**
    Update the inner function value estimators

$$u_{i,t}^{(1)} = (1-\gamma_t)u_{i,t-1}^{(1)} + \gamma_t \frac{1}{B_1+B_2} \sum_{\mathbf{x}_j \in |\mathcal{B}_t^+ \cup \mathcal{B}_t^-|} \mathbb{I}(y_j = 1)\ell(h(\mathbf{w}_t;\mathbf{x}_j) - h(\mathbf{w}_t;\mathbf{x}_i)),$$

$$u_{i,t}^{(2)} = (1-\gamma_t)u_{i,t-1}^{(2)} + \gamma_t \frac{1}{B_1+B_2} \sum_{\mathbf{x}_j \in |\mathcal{B}_t^+ \cup \mathcal{B}_t^-|} \ell(h(\mathbf{w}_t;\mathbf{x}_j) - h(\mathbf{w}_t;\mathbf{x}_i)),$$

  end for
  Set $\mathbf{u}_{i,t} = \mathbf{u}_{i,t-1}, i \notin \mathcal{B}_t^+$
  Compute the vanilla gradient estimator

$$\mathbf{z}_t = \frac{1}{B_1} \sum_{\mathbf{x}_i \in \mathcal{B}_t^+} \frac{1}{B_1+B_2} \sum_{\mathbf{x}_j \in |\mathcal{B}_t^+ \cup \mathcal{B}_t^-|} \frac{u_{i,t}^{(1)} - u_{i,t}^{(2)}\mathbb{I}(y_j = 1)}{(u_{i,t}^{(2)})^2} \nabla \ell(h(\mathbf{w}_t;\mathbf{x}_j) - h(\mathbf{w}_t;\mathbf{x}_i))$$

  Update $\mathbf{w}_{t+1}$ by Momentum or AdamW
end for

the initial update of $\mathbf{u}_{i,t}$ coincides with the mini-batch estimate of the inner function at that point. This technique will be used in other FCCO applications.

Feature Learning

Similar to AUC maximization, the end-to-end training based on the AP surrogate objective does not favor feature learning. To mitigate this issue, one can first pretrain the encoder network on the training data by traditional supervised learning (e.g. ERM with the CE loss) or self-supervised representation learning and then fine-tune the feature extraction layers and a random initialized classifier layer by optimizing an AP surrogate objective. The compositional training could be also employed for unified feature learning and AP maximization.

Moving-average parameter γ_t

In practice, we can set $\gamma_t = \gamma$ and tune γ in the range $(0, 1)$ to optimize the validation performance.

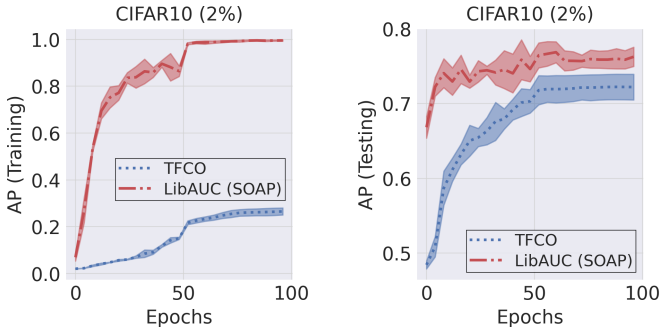


Fig. 6.11: Comparison of different methods for AP maximization. TFCO refers to the constrained optimization algorithm implemented in the Google TensorFlow Constrained Optimization library. The experiment was conducted on a constructed imbalanced binary classification task of CIFAR10, which originally contains 10 classes. These classes are partitioned into two equal groups to form the positive and negative classes based on their class IDs. The test data is unchanged (i.e., the testing data is still balanced). For more details, please refer to (Yuan et al., 2023b).

6.4.3 Stochastic Partial AUC Maximization

Stochastic OPAUC Maximization

We focus on maximizing the OPAUC with the false positive rate (FPR) restricted to the range $[0, \beta]$. As shown in Section 2.3.3, OPAUC maximization can be formulated as minimizing a surrogate objective:

$$\min_{\mathbf{w}} \frac{1}{n_+}\frac{1}{k} \sum_{\mathbf{x}_i \in \mathcal{S}_+} \sum_{\mathbf{x}_j \in \mathcal{S}_-^{\downarrow}[1,k]} \ell(h(\mathbf{w}; \mathbf{x}_j) - h(\mathbf{w}; \mathbf{x}_i)), \tag{6.23}$$

where $k = \lfloor n_- \beta \rfloor$, $\mathcal{S}^{\downarrow}[1, k] \subseteq \mathcal{S}$ denotes the subset of examples whose rank in terms of their prediction scores in the descending order are in the range of $[1, k]$, and $\ell(\cdot)$ denotes a continuous surrogate pairwise loss such as in Table 2.3.

The challenge lies at how to tackle the top-k selection $\mathbf{x}_j \in \mathcal{S}_-^{\downarrow}[1, k]$. Below, we present two approaches: a direct approach that leverages the dual form of CVaR and an indirect approach that replaces the top-k selection by soft weighting.

A Direct Approach

This approach will be restricted to a non-decreasing pairwise loss function $\ell(s)$. Under this assumption, the ranking over negative samples by their prediction scores $h(\mathbf{w}; \mathbf{x}_j)$ is equivalent to that by the pairwise loss $\ell(h(\mathbf{w}; \mathbf{x}_j) - h(\mathbf{w}; \mathbf{x}_i)), \mathbf{x}_j \in \mathcal{S}_i$. Hence, the average of pairwise losses over top-k negatives

Algorithm 29 SOPA for solving (6.26) of direct OPAUC maximization

1: Initialize $\mathbf{w}$ and $\boldsymbol{\nu}_1 = 0$
2: **for** $t = 1, \ldots, T$ **do**
3: Draw B_1 positive data $\mathcal{B}_t^+ \subset \mathcal{S}_+$ and B_2 negative data $\mathcal{B}_t^- \subset \mathcal{S}_-$
4: Compute $p_{ij} = \mathbb{I}(\ell(h(\mathbf{w}_t, \mathbf{x}_j) - h(\mathbf{w}_t, \mathbf{x}_i)) - \nu_{i,t} > 0)$ for $\mathbf{x}_i \in \mathcal{B}_t^+, \mathbf{x}_j \in \mathcal{B}_t^-$
5: **for** $i \in \mathcal{B}_t^+$ **do**
6: Update $\nu_{i,t+1} = \nu_{i,t} - \eta_2(\frac{k}{n_-} - \frac{1}{B_2}\sum_{\mathbf{x}_j \in \mathcal{B}_t^-} p_{ij})$
7: **end for**
8: Set $\nu_{i,t+1} = \nu_{i,t}, i \notin \mathcal{B}_t^+$
9: Compute a vanilla gradient estimator $\mathbf{z}_t$ by

$$\mathbf{z}_t = \frac{1}{B_1 B_2} \sum_{\mathbf{x}_i \in \mathcal{B}_t^+} \sum_{\mathbf{x}_j \in \mathcal{B}_t^-} p_{ij} \nabla_{\mathbf{w}} \ell(h(\mathbf{w}_t, \mathbf{x}_j) - h(\mathbf{w}_t, \mathbf{x}_i))$$

10: Update $\mathbf{w}_{t+1}$ by SGD or Momentum or AdamW
11: **end for**

$$L_i(\mathbf{w}) = \frac{1}{k} \sum_{\mathbf{x}_j \in \mathcal{S}_-^{\downarrow}[1,k]} \ell(h(\mathbf{w}; \mathbf{x}_j) - h(\mathbf{w}; \mathbf{x}_i)) \tag{6.24}$$

is equivalent to the average of top-k pairwise losses over negative data, i.e., an empirical CVaR estimator. Then leveraging the dual form of CVaR (2.15), we transform the above loss into a minimization problem, i.e.,

$$L_i(\mathbf{w}) = \min_{\nu_i} \frac{1}{k} \sum_{\mathbf{x}_j \in \mathcal{S}_-} [\ell(h(\mathbf{w}; \mathbf{x}_j) - h(\mathbf{w}; \mathbf{x}_i)) - \nu_i]_+ + \nu_i. \tag{6.25}$$

As a result, we have the following equivalent reformulation.

Lemma 6.1 (Reformulation of OPAUC maximization.) *When $\ell(\cdot)$ is non-decreasing, then the problem (6.23) for OPAUC maximization is equivalent to*

$$\min_{\mathbf{w}, \boldsymbol{\nu} \in \mathbb{R}^{n_+}} F(\mathbf{w}, \boldsymbol{\nu}) = \frac{1}{n_+} \sum_{\mathbf{x}_i \in \mathcal{S}_+} \left\{ \frac{k}{n_-} \nu_i + \frac{1}{n_-} \sum_{\mathbf{x}_j \in \mathcal{S}_-} (\ell(h(\mathbf{w}, \mathbf{x}_j) - h(\mathbf{w}, \mathbf{x}_i)) - \nu_i)_+ \right\}, \tag{6.26}$$

The above problem is a special case of compositional OCE studied in Section 5.5.

A benefit for solving (6.26) is that an unbiased stochastic subgradient can be computed in terms of $(\mathbf{w}, \boldsymbol{\nu})$. We present a method in Algorithm 29, which is an application of the ASGD and is referred to as SOPA. A key feature of SOPA is that the stochastic gradient estimator for $\mathbf{w}$ (Step 9) is a weighted average gradient of the pairwise losses for all pairs in the mini-batch. The weights p_{ij} (either 0 or 1) are dynamically computed by Step 4, which compares the pairwise loss $(\ell(h(\mathbf{w}_t, \mathbf{x}_i) - h(\mathbf{w}_t, \mathbf{x}_j))$ with the threshold variable $\nu_{i,t}$, which is also updated by an SGD step.

Algorithm 30 SOPA-s for solving (6.28) of indirect OPAUC maximization

1: Initialize $\mathbf{w}, \mathbf{u}_0$
2: **for** $t = 1, \ldots, T$ **do**
3: Draw B_1 positive data $\mathcal{B}_t^+ \subset \mathcal{S}_+$ and B_2 negative data $\mathcal{B}_t^- \subset \mathcal{S}_-$
4: **for** $i \in \mathcal{B}_t^+$ **do**
5: Update $u_{i,t} = (1-\gamma)u_{i,t-1} + \gamma \frac{1}{B_2} \sum_{\mathbf{x}_j \in \mathcal{B}_t^-} \exp\left(\frac{\ell(h(\mathbf{w}_t;\mathbf{x}_j) - h(\mathbf{w}_t;\mathbf{x}_i))}{\tau}\right)$
6: **end for**
7: Set $u_{i,t} = u_{i,t-1}, i \notin \mathcal{B}_t^+$
8: Compute $p_{ij} = \exp(\ell(h(\mathbf{w}_t;\mathbf{x}_j) - h(\mathbf{w}_t;\mathbf{x}_i))/\tau)/u_{i,t}$ for $\mathbf{x}_i \in \mathcal{B}_t^+, \mathbf{x}_j \in \mathcal{B}_t^-$
9: Compute a vanilla gradient estimator $\mathbf{z}_t$ by

$$\mathbf{z}_t = \frac{1}{B_1 B_2} \sum_{\mathbf{x}_i \in \mathcal{B}_t^+} \sum_{\mathbf{x}_j \in \mathcal{B}_t^-} p_{ij} \nabla \ell(h(\mathbf{w}_t;\mathbf{x}_j) - h(\mathbf{w}_t;\mathbf{x}_i))$$

10: Update $\mathbf{w}_{t+1}$ by Momentum or AdamW method.
11: **end for**

The convergence guarantee of SOPA using the SGD update for $\mathbf{w}_t$ has been established in Section 5.5. In practice, the convergence speed of SOPA may be further accelerated by integrating Momentum or Adam updates for the model parameter $\mathbf{w}$.

An indirect approach by FCCO

Due to the connection between CVaR and DRO (2.13), an alternative approach is to replace the top-k pairwise loss $L_i(\mathbf{w})$ by a KL-regularized DRO, i.e.,

$$\begin{aligned}\hat{L}_i(\mathbf{w}) &= \max_{\mathbf{p} \in \Delta_n} \sum_{\mathbf{x}_j \in \mathcal{S}_-} p_j \ell(h(\mathbf{w};\mathbf{x}_j) - h(\mathbf{w};\mathbf{x}_i)) - \tau \mathrm{KL}(\mathbf{p}, 1/n_-) \\ &= \tau \log\left(\frac{1}{n_-} \sum_{\mathbf{x}_j \in \mathcal{S}_-} \exp\left(\frac{\ell(h(\mathbf{w};\mathbf{x}_j) - h(\mathbf{w};\mathbf{x}_i))}{\tau}\right)\right).\end{aligned} \tag{6.27}$$

As a result, an indirect approach for OPAUC maximization is to solve the following FCCO problem:

$$\min_{\mathbf{w}} \frac{1}{n_+} \sum_{i=1}^{n_+} \tau \log\left(\frac{1}{n_-} \sum_{\mathbf{x}_j \in \mathcal{S}_-} \exp\left(\frac{\ell(h(\mathbf{w};\mathbf{x}_j) - h(\mathbf{w};\mathbf{x}_i))}{\tau}\right)\right). \tag{6.28}$$

An application of the SOX algorithm is given in Algorithm 30, which is referred to as SOPA-s. The key difference between SOPA-s and SOPA lies at the pairwise weights p_{ij} in SOPA-s (Step 8) are soft weights between 0 and 1, in contrast to the hard weights $p_{ij} \in \{0, 1\}$ in SOPA.

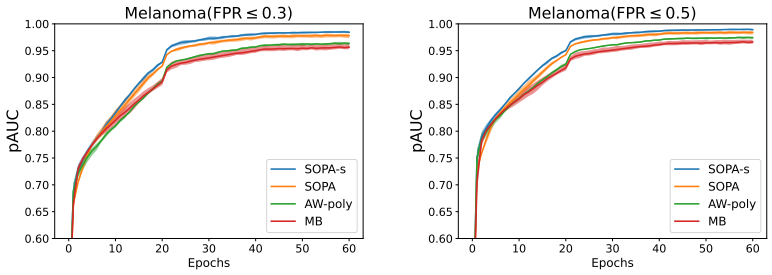


Fig. 6.12: Comparison of different methods for OPAUC maximization with FPR less than $\beta = 0.3$ (left) and $\beta = 0.5$ (right). The dataset is Melanoma classification from Kaggle competition. The training set has only 1.76% positive (malignant) samples. MB refers to the BSGD approach that computes gradients using only the top β% of negative examples within each mini-batch; AW-Poly is a heuristic weighted method that assigns weights to negative samples in the mini-batch using a manually designed weighting function. For more details, please refer to (Zhu et al., 2022b).

Stochastic TPAUC Maximization

As shown in Section 2.3.3, empirical maximization of TPAUC with FPR $\leq \beta$, TPR $\geq \alpha$ can be formulated as:

$$\min_{\mathbf{w}} \frac{1}{k_1}\frac{1}{k_2} \sum_{\mathbf{x}_i \in \mathcal{S}_+^{\uparrow}[1,k_1]} \sum_{\mathbf{x}_j \in \mathcal{S}_-^{\downarrow}[1,k_2]} \ell(h(\mathbf{w};\mathbf{x}_j) - h(\mathbf{w};\mathbf{x}_i)), \tag{6.29}$$

where $k_1 = \lfloor n_+(1-\alpha) \rfloor$, $k_2 = \lfloor n_-\beta \rfloor$. If we define

$$L_i(\mathbf{w}) = \frac{1}{k_2} \sum_{\mathbf{x}_j \in \mathcal{S}_-^{\downarrow}[1,k_2]} \ell(h(\mathbf{w};\mathbf{x}_j) - h(\mathbf{w};\mathbf{x}_i)), \tag{6.30}$$

then, the problem in (6.29) can be written as:

$$\min_{\mathbf{w}} \frac{1}{k_1} \sum_{\mathbf{x}_i \in \mathcal{S}_+^{\uparrow}[1,k_1]} L_i(\mathbf{w}). \tag{6.31}$$

Similar to OPAUC maximization, we will present a direct approach and an indirect approach.

A Direct Approach

The first approach is based on the following reformulation of TPAUC maximization.

Algorithm 31 STACO for solving (6.32) of direct TPAUC maximization

Initialize $\mathbf{w}$ and $\nu_1 = 0$, $\nu' = 0$
for $t = 1, \ldots, T$ **do**
  Draw B_1 positive data $\mathcal{B}_t^+ \subset \mathcal{S}_+$ and B_2 negative data $\mathcal{B}_t^- \subset \mathcal{S}_-$
  Compute $p_{ij} = \mathbb{I}(\ell(h(\mathbf{w}_t, \mathbf{x}_i) - h(\mathbf{w}_t, \mathbf{x}_j)) - \nu_{i,t} > 0)$ for $\mathbf{x}_i \in \mathcal{B}_t^+, \mathbf{x}_j \in \mathcal{B}_t^-$
  for $i \in \mathcal{B}_t^+$ **do**
    Update $y_{i,t+1}$ and $\nu_{i,t+1}$ by

$$y_{i,t+1} = \left[y_{i,t} - \eta_2 \left\{ \frac{1}{B_2} \sum_{\mathbf{x}_j \in \mathcal{B}_t^-} (\ell(h(\mathbf{w}_t; \mathbf{x}_j) - h(\mathbf{w}_t; \mathbf{x}_i)) - \nu_{i,t})_+ + \frac{k_2}{n_-}(\nu_{i,t} - \nu'_t) \right\} \right]_{[0,1]}$$

$$\nu_{i,t+1} = \nu_{i,t} - \eta_1 y_{i,t+1} \left(\frac{k_2}{n_-} - \frac{1}{B_2} \sum_{\mathbf{x}_j \in \mathcal{B}_t^-} p_{ij} \right)$$

  end for
  Set $y_{i,t+1} = y_{i,t}, i \notin \mathcal{B}_t^+$ and $\nu_{i,t+1} = \nu_{i,t}, t \notin \mathcal{B}_t^+$
  Update $\nu'_{t+1} = \nu'_t - \eta_1 (\frac{k_1 k_2}{n_+ n_-} - \frac{k_2}{n_-} \frac{1}{B_1} \sum_{\mathbf{x}_i \in \mathcal{B}_t^+} y_{i,t+1})$
  Compute a vanilla gradient estimator $\mathbf{z}_t$ by

$$\mathbf{z}_t = \frac{1}{B_1 B_2} \sum_{\mathbf{x}_i \in \mathcal{B}_t^+} \sum_{\mathbf{x}_j \in \mathcal{B}_t^-} y_{i,t+1} p_{ij} \nabla_{\mathbf{w}} \ell(h(\mathbf{w}_t, \mathbf{x}_j) - h(\mathbf{w}_t, \mathbf{x}_i))$$

  Update $\mathbf{w}_{t+1}$ by SGD, Momentum or AdamW
end for

Lemma 6.2 (Reformulation of TPAUC maximization.) *When $\ell(\cdot)$ is non-decreasing, the problem (6.29) for TPAUC maximization is equivalent to*

$$\min_{\mathbf{w}, \boldsymbol{\nu}, \nu'} \frac{1}{n_+} \sum_{\mathbf{x}_i \in \mathcal{S}_+} f(g_i(\mathbf{w}, \boldsymbol{\nu}, \nu')) + \frac{k_1 k_2}{n_+ n_-} \nu', \tag{6.32}$$

where $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_{n_+})^\top$, $f(\cdot) = [\cdot]_+$ and

$$g_i(\mathbf{w}, \boldsymbol{\nu}, \nu') = \frac{1}{n_-} \sum_{\mathbf{x}_j \in \mathcal{S}_-} (\ell(h(\mathbf{w}; \mathbf{x}_j) - h(\mathbf{w}; \mathbf{x}_i)) - \nu_i)_+ + \frac{k_2}{n_-}(\nu_i - \nu').$$

We leave the proof as an excise for the reader.

It is clear that the problem (6.32) is an instance of FCCO, where the outer function is non-smooth and monotonically non-decreasing. Hence, SONX, SONEX, and ALEXR can be applied. We present an application of ALEXR for solving the above problem in Algorithm 31 (referred to as STACO) based on its min-max reformulation:

Algorithm 32 SOTA-s for solving (6.33) of Indirect TPAUC Maximization

Initialize $\mathbf{w}_1, \mathbf{u}_1, v_1$,
for $t = 1, \ldots, T$ **do**
  Draw B_1 positive data $\mathcal{B}_t^+ \subset \mathcal{S}_+$ and B_2 negative data $\mathcal{B}_t^- \subset \mathcal{S}_-$
  for $i \in \mathcal{B}_t^+$ **do**
    Update $u_{i,t} = (1-\gamma_0)u_{i,t-1} + \gamma_0 \frac{1}{B_2} \sum_{\mathbf{x}_j \in \mathcal{B}_-^t} \exp\left(\frac{\ell(h(\mathbf{w}_t;\mathbf{x}_j) - h(\mathbf{w}_t;\mathbf{x}_i))}{\tau_2}\right)$
  end for
  Set $u_{i,t} = u_{i,t-1}, i \notin \mathcal{B}_t^+$
  Let $v_t = (1-\gamma_1)v_{t-1} + \gamma_1 \frac{1}{B_1} \sum_{\mathbf{x}_i \in \mathcal{B}_t^+} (u_{i,t})^{\tau_2/\tau_1}$
  Compute

$$p_{ij} = \frac{\exp(\ell(h(\mathbf{w}_t;\mathbf{x}_j) - h(\mathbf{w}_t;\mathbf{x}_i))/\tau_2)(u_{i,t})^{\tau_2/\tau_1 - 1}}{v_t}, \forall \mathbf{x}_i \in \mathcal{B}_t^+, \mathbf{x}_j \in \mathcal{B}_t^-$$

  Compute a vanilla gradient estimator $\mathbf{z}_t$ by

$$\mathbf{z}_t = \frac{1}{B_1 B_2} \sum_{\mathbf{x}_i \in \mathcal{B}_t^+} \sum_{\mathbf{x}_j \in \mathcal{B}_t^-} p_{ij} \nabla \ell(h(\mathbf{w}_t;\mathbf{x}_j) - h(\mathbf{w}_t;\mathbf{x}_i))$$

  Update $\mathbf{w}_{t+1}$ by Momentum or AdamW
end for

$$\min_{\mathbf{w},\nu,\nu'} \max_{\mathbf{y}\in[0,1]^{n_+}} \frac{1}{n_+} \sum_{\mathbf{x}_i \in \mathcal{S}_+} y_i \left[\frac{1}{n_-} \sum_{\mathbf{x}_j \in \mathcal{S}_-} (\ell(\mathbf{w};\mathbf{x}_i,\mathbf{x}_j) - \nu_i)_+ + \frac{k_2}{n_-}(\nu_i - \nu') \right] + \frac{k_1 k_2}{n_+ n_-} \nu'.$$

An Indirect Approach

Following the strategy used in OPAUC maximization, we adopt an indirect approach by replacing top-k estimators with their KL-regularized DRO counterparts, which yield smooth surrogate objectives.

With a non-decreasing pairwise surrogate loss $\ell(\cdot)$, $L_i(\mathbf{w})$ is a non-increasing function of $h(\mathbf{w};\mathbf{x}_i)$, the average of $L_i(\mathbf{w})$ over bottom-k_1 positive examples in (6.31) is equivalent to the average of top-k_1 losses $L_i(\mathbf{w})$ over all positive data. Hence, we approximate the resulting top-k_1 estimator by a KL-regularized objective:

$$\tau_1 \log\left(\frac{1}{n_+} \sum_{\mathbf{x}_i \in \mathcal{S}_+} \exp\left(\frac{L_i(\mathbf{w})}{\tau_1}\right)\right).$$

Then, we substitute $L_i(\mathbf{w})$ with $\hat{L}_i(\mathbf{w})$ as defined in (6.27), leading to the following smoothed objective:

$$
\begin{aligned}
F(\mathbf{w}) &= \tau_1 \log\left(\frac{1}{n_+}\sum_{\mathbf{x}_i\in\mathcal{S}_+}\exp\left(\frac{\hat{L}_i(\mathbf{w})}{\tau_1}\right)\right) \\
&= \tau_1 \log\left(\frac{1}{n_+}\sum_{\mathbf{x}_i\in\mathcal{S}_+}\exp\left(\frac{\tau_2\log\left(\frac{1}{n_-}\sum_{\mathbf{x}_j\in\mathcal{S}_-}\exp\left(\frac{\ell(h(\mathbf{w};\mathbf{x}_j)-h(\mathbf{w};\mathbf{x}_i))}{\tau_2}\right)\right)}{\tau_1}\right)\right) \\
&= \tau_1 \log\left(\frac{1}{n_+}\sum_{\mathbf{x}_i\in\mathcal{S}_+}\left(\frac{1}{n_-}\sum_{\mathbf{x}_j\in\mathcal{S}_-}\exp\left(\frac{\ell(h(\mathbf{w};\mathbf{x}_j)-h(\mathbf{w};\mathbf{x}_i))}{\tau_2}\right)\right)^{\frac{\tau_2}{\tau_1}}\right).
\end{aligned}
$$

To minimize this objective, we formulate the problem as a three-level compositional stochastic optimization:

$$
\min_{\mathbf{w}} f_1\left(\frac{1}{n_+}\sum_{\mathbf{x}_i\in\mathcal{S}_+} f_2(g_i(\mathbf{w}))\right), \tag{6.33}
$$

where $f_1(s) = \tau_1\log(s)$, $f_2(g) = g^{\tau_2/\tau_1}$, and

$$
g_i(\mathbf{w}) = \frac{1}{n_-}\sum_{\mathbf{x}_j\in\mathcal{S}_-}\exp\left(\frac{\ell(h(\mathbf{w};\mathbf{x}_j)-h(\mathbf{w};\mathbf{x}_i))}{\tau_2}\right).
$$

The inner function of f_1 exhibits a finite-sum coupled compositional optimization (FCCO) structure. To accurately estimate $\nabla f_1(\cdot)$ at the inner function value, we maintain a moving average estimator v_t to track $\frac{1}{n_+}\sum_{\mathbf{x}_i\in\mathcal{S}_+} f_2(g_i(\mathbf{w}_t))$.

We present a stochastic optimization algorithm—referred to as SOTA-s—for solving this problem in Algorithm 32. We update $u_{i,t}$ to track $g_i(\mathbf{w}_t)$ in Step 5 and maintain v_t to estimate $\frac{1}{n_+}\sum_{\mathbf{x}_i\in\mathcal{S}_+} f_2(g_i(\mathbf{w}_t))$ in Step 8. The gradient estimator in Step 9 is given by:

$$
\nabla f_1(v_t)\cdot\frac{1}{|\mathcal{B}_+|}\sum_{\mathbf{x}_i\in\mathcal{B}_t^+}\nabla f_2(u_{i,t})\cdot\nabla\hat{g}_i(\mathbf{w}_t),
$$

where $\hat{g}_i(\mathbf{w}_t) = \frac{1}{B_2}\sum_{\mathbf{x}_j\sim\mathcal{B}_t^-}\exp\left(\frac{\ell(h(\mathbf{w}_t;\mathbf{x}_j)-h(\mathbf{w}_t;\mathbf{x}_i))}{\tau_2}\right)$.

6.4.4 Stochastic NDCG Maximization

In Section 2.3.4, we have formulated NDCG maximization as the following empirical X-risk minimization problem:

$$
\min_{\mathbf{w}}\frac{1}{N}\sum_{q=1}^{N}\frac{1}{Z_q}\sum_{\mathbf{x}_{q,i}\in\mathcal{S}_q^+}\frac{1-2^{y_{q,i}}}{\log_2(N_q g(\mathbf{w};\mathbf{x}_{q,i},\mathcal{S}_q)+1)}, \tag{6.34}
$$

Algorithm 33 SONG

Initialize $\mathbf{w}_1, \mathbf{u}_0$
for $t = 1, \ldots T$ **do**
  Draw some relevant Q-I pairs $\mathcal{B}_t = \{(q, \mathbf{x}_{q,i})\} \subset \mathcal{S}$
  For each sampled q draw a batch of items $\mathcal{B}_q^t \subset \mathcal{S}_q$
  for each sampled Q-I pair $(q, \mathbf{x}_{q,i}) \in \mathcal{B}_t$ **do**
    Compute $u_{q,i,t} = (1-\gamma)u_{q,i,t-1} + \gamma \frac{1}{|\mathcal{B}_q^t|} \sum_{\mathbf{x}' \in \mathcal{B}_q^t} \ell(s(\mathbf{w}_t; \mathbf{x}', q) - s(\mathbf{w}_t; \mathbf{x}_{q,i}, q))$
    Compute

$$p_{q,i} = \nabla f_{q,i}(u_{q,i,t}) = \frac{(2^{y_{q,i}} - 1)N_q}{Z_q(N_q u_{q,i,t} + 1)\log_2^2(N_q u_{q,i,t} + 1)\ln(2)}$$

  end for
  Compute a vanilla gradient estimator $\mathbf{z}_t$ by

$$\mathbf{z}_t = \frac{1}{|\mathcal{B}_t|} \sum_{(q,\mathbf{x}_{q,i}) \in \mathcal{B}_t} p_{q,i} \frac{1}{|\mathcal{B}_q^t|} \sum_{\mathbf{x}' \in \mathcal{B}_q^t} \ell(s(\mathbf{w}; \mathbf{x}', q) - s(\mathbf{w}; \mathbf{x}, q))$$

  update $\mathbf{w}_{t+1}$ by Momentum and AdamW optimizer
end for

where $N_q g(\mathbf{w}; \mathbf{x}, \mathcal{S}_q) = \sum_{\mathbf{x}' \in \mathcal{S}_q} \ell(s(\mathbf{w}; \mathbf{x}', q) - s(\mathbf{w}; \mathbf{x}, q))$ is a surrogate of the rank function $r(\mathbf{w}; \mathbf{x}, S_q) = \sum_{\mathbf{x}' \in \mathcal{S}_q} \mathbb{I}(s(\mathbf{w}; \mathbf{x}', q) - s(\mathbf{w}; \mathbf{x}, q) \geq 0)$, and $s(\mathbf{w}; \mathbf{x}, q)$ denotes the predicted relevance score for item $\mathbf{x}$ with respect to query q, parameterized by $\mathbf{w} \in \mathbb{R}^d$ (e.g., a deep neural network).

As a result, NDCG maximization can be rewritten as an instance of FCCO:

$$\min_{\mathbf{w} \in \mathbb{R}^d} \frac{1}{|\mathcal{S}|} \sum_{(q,\mathbf{x}_{q,i}) \in \mathcal{S}} f_{q,i}(g(\mathbf{w}; \mathbf{x}_{q,i}, \mathcal{S}_q)), \tag{6.35}$$

where $\mathcal{S} = \{(q, \mathbf{x}_{q,i}) \mid q \in \mathcal{Q}, \mathbf{x}_{q,i} \in \mathcal{S}_q^+\}$ represent the collection of all relevant query-item (Q-I) pairs, and

$$f_{q,i}(g) = \frac{1}{Z_q} \frac{1 - 2^{y_{q,i}}}{\log_2(N_q g + 1)}.$$

We apply the SOX method to this problem as shown in Algorithm 33, which we call SONG.

Top-K NDCG Maximization

In practice, top-K NDCG is the preferred metric for information retrieval and recommender systems, as users primarily focus on the highest-ranked items. It is defined as:

$$\frac{1}{N}\sum_{q=1}^{N}\frac{1}{Z_q^{(K)}}\sum_{\mathbf{x}_{q,i}\in\mathcal{S}_q^+}\mathbb{I}(\mathbf{x}_{q,i}\in\mathcal{S}_q^{(K)})\cdot\frac{2^{y_{q,i}}-1}{\log_2(r(\mathbf{w};\mathbf{x}_{q,i},\mathcal{S}_q)+1)},$$

where $\mathcal{S}_q^{(K)}$ is the set of top-K items based on predicted scores, and $Z_q^{(K)}$ is the ideal DCG in the top-K positions.

Optimizing top-K NDCG introduces an added complexity: selecting the top-K items is non-differentiable. Unlike pAUC, where a top-K estimator exists, the surrogate function

$$\frac{2^{y_{q,i}}-1}{\log_2(N_q g(\mathbf{w};\mathbf{x}_{q,i},\mathcal{S}_q)+1)}$$

is not generally monotonic in the score $s(\mathbf{w};\mathbf{x}_{q,i},q)$ unless all $y_{q,i}$ values are identical. We consider two approaches to handle this problem.

Approach 1: Surrogate for Top-K Inclusion

We use the identity $\mathbb{I}(\mathbf{x}_{q,i}\in\mathcal{S}_q^{(K)})=\mathbb{I}(K-r(\mathbf{w};\mathbf{x}_{q,i},\mathcal{S}_q)\geq 0)$ and approximate it by a non-decreasing surrogate $\psi(K-N_q g(\mathbf{w};\mathbf{x}_{q,i},\mathcal{S}_q))$, e.g., the sigmoid function. The resulting objective becomes:

$$\min_{\mathbf{w}\in\mathbb{R}^d}\frac{1}{|\mathcal{S}|}\sum_{q=1}^{N}\sum_{\mathbf{x}_{q,i}\in\mathcal{S}_q^+}\psi(K-N_q g(\mathbf{w};\mathbf{x}_{q,i},\mathcal{S}_q))\cdot\frac{1-2^{y_{q,i}}}{Z_q^{(K)}\log_2(N_q g(\mathbf{w};\mathbf{x}_{q,i},\mathcal{S}_q)+1)}. \tag{6.36}$$

This can be optimized using FCCO techniques.

Approach 2: Threshold Estimation via Bilevel Optimization

Denote by $\lambda_q(\mathbf{w})$ the the $(K+1)$-th largest score among all $\mathbf{x}'\in\mathcal{S}_q$. We use the identity $\mathbb{I}(\mathbf{x}_{q,i}\in\mathcal{S}_q^{(K)})=\mathbb{I}(s(\mathbf{w};\mathbf{x}_{q,i},q)>\lambda_q(\mathbf{w}))$ and approximate it by $\psi(s(\mathbf{w};\mathbf{x}_{q,i},q)-\lambda_q(\mathbf{w}))$. The threshold $\lambda_q(\mathbf{w})$ can be computed by solving a convex optimization problem as shown in the lemma below.

Lemma 6.3 *Let* $\lambda_q(\mathbf{w})=\arg\min_\lambda(K+\varepsilon)\lambda+\sum_{\mathbf{x}'\in\mathcal{S}_q}(s(\mathbf{w};\mathbf{x}',q)-\lambda)_+$ *for any* $\varepsilon\in(0,1)$*, then* $\lambda_q(\mathbf{w})$ *is the* $(K+1)$*-th largest value among* $\{s(\mathbf{w};\mathbf{x}',q)|\mathbf{x}'\in\mathcal{S}_q\}$.

As a result, we formulate the following bilevel optimization problem for top-K NDCG maximization:

$$\begin{aligned}\min_{\mathbf{w}}\quad & \frac{1}{|\mathcal{S}|}\sum_{q=1}^{N}\sum_{\mathbf{x}_{q,i}\in\mathcal{S}_q^+}\frac{\psi(s(\mathbf{w};\mathbf{x}_{q,i},q)-\lambda_q(\mathbf{w}))\cdot(1-2^{y_{q,i}})}{Z_q^{(K)}\log_2(N_q g(\mathbf{w};\mathbf{x}_{q,i},\mathcal{S}_q)+1)}\\ \text{s.t.}\quad & \lambda_q(\mathbf{w})=\arg\min_\lambda\frac{K+\varepsilon}{N_q}\lambda+\frac{1}{N_q}\sum_{\mathbf{x}'\in\mathcal{S}_q}(s(\mathbf{w};\mathbf{x}',q)-\lambda)_+,\quad\forall q.\end{aligned} \tag{6.37}$$

This bilevel formulation is challenging due to the non-smooth and non-strongly-convex lower-level problem. One remedy is to apply Nesterov smoothing to the hinge loss (see Example 5.1) and add a small quadratic regularization term of λ to the lower level objective. This allows employing the Approach 1 of using moving-average estimators from Section 4.5.3.

In practice, we can ignore the gradient of ψ and adapt the SONG algorithm by updating λ_q iteratively and modifying $p_{q,i}$ as:

$$\lambda_{q,t+1} = \lambda_{q,t} - \eta' \left(\frac{K+\varepsilon}{N_q} + \frac{1}{|\mathcal{B}_q|} \sum_{\mathbf{x}' \in \mathcal{B}_q^t} \mathbb{I}(s(\mathbf{w}_t; \mathbf{x}', q) > \lambda) \right), \quad \forall q \in \mathcal{B}_t,$$
$$p_{q,i} = \psi(s(\mathbf{w}_t; \mathbf{x}_{q,i}, q) - \lambda_{q,t+1}) \cdot \nabla f_{q,i}(u_{q,i,t}).$$

As with other non-decomposable metrics, it is beneficial to first pretrain the model by optimizing the listwise cross-entropy loss, which itself is an FCCO problem, as defined in (2.47).

6.4.5 The LibAUC Library

The algorithms presented in Section 6.4 for various X-risk minimization tasks share several common features: (1) they all require sampling both positive and negative examples; (2) their vanilla gradient updates involve a weighted sum of gradients from pairwise losses computed on the sampled data; and (3) they utilize moving-average estimators to track inner function values. These shared characteristics motivate the design of a unified implementation pipeline. To this end, the `LibAUC` library was developed to encapsulate these principles within a modular and extensible framework, built on top of the PyTorch ecosystem. Below, we highlight several key components of `LibAUC`. For tutorials and source code, we refer interested readers to the GitHub repository:

LibAUC GitHub Repository

`https://github.com/Optimization-AI/LibAUC`

Pipeline

The training pipeline of a deep neural network in the `LibAUC` library is illustrated in Figure 6.13. It consists of five core modules: `Dataset`, `Controlled Data Sampler`, `Model`, `Dynamic Mini-batch Loss`, and `Optimizer`. While the `Dataset`, `Model`, and `Optimizer` modules align closely with those in standard training frameworks, the key innovations lie in the `Dynamic Mini-batch Loss` and `Controlled Data Sampler` modules.

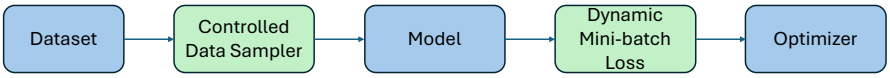


Fig. 6.13: Training pipeline of the LibAUAC library for deep learning.

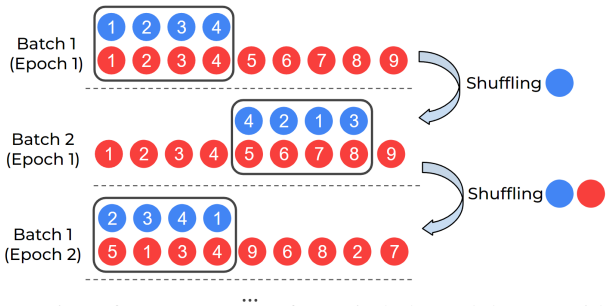


Fig. 6.14: Illustration of `DualSampler` for an imbalanced dataset with 4 positives ● and 9 negatives ●.

The `Dynamic Mini-batch Loss` module defines the loss using dynamically updated variables, which are computed and refined with forward propagation results. This design ensures that compositional gradients can be correctly estimated from mini-batch samples using backpropagation. The `Controlled Data Sampler` module, in contrast to standard random sampling strategies, allows fine-grained control over the ratio of positive to negative samples. This control can be tuned to improve learning effectiveness and overall performance.

Controlled Data Sampler

Unlike traditional ERM, EXM requires sampling to estimate the outer average and the inner average. In algorithms for AUC, AP, OPAUC and TPAUC optimization, we need to sample two mini-batches $\mathcal{B}_+^t \subset \mathcal{S}_+$ and $\mathcal{B}_-^t \subset \mathcal{S}_-$ at each iteration t. When the total batch size is fixed, balancing the mini-batch size for outer average and that for the inner average could be beneficial for accelerating convergence according to our theoretical analysis in Chapter 5. Hence, the `Controlled Data Sampler` module can help ensure that both positive and negative samples will be sampled and the proportion of positive samples in the mini-batch can be controlled by a hyper-parameter.

DualSampler. For binary classification problems, DualSampler takes as input hyper-parameters such as `batch_size` and `sampling_rate`, and generates the customized mini-batch samples, where `sampling_rate` controls the number of positive samples in the mini-batch according to the formula:

$$\texttt{\#positives} = \texttt{batch\_size} * \texttt{sampling\_rate}.$$

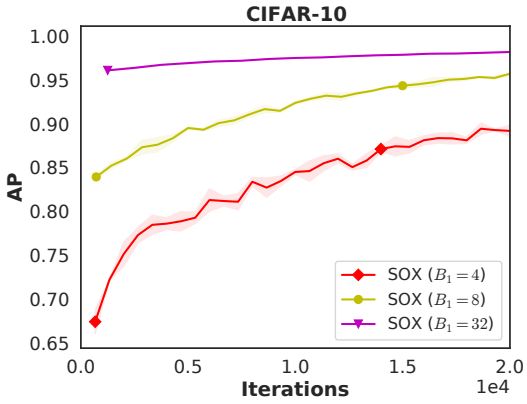


Fig. 6.15: The training curves of AP for different number of positive examples per mini-batch in DualSampler when the total batch size is fixed to 64. The algorithm is SOPA - a variant of SOX. Experiments were conducted on a constructed imbalanced binary classification task derived from CIFAR-10, identical to the setting used in Figure 6.11.

Figure 6.14 shows an example of `DualSampler` for constructing mini-batch data with even positive and negative samples on an imbalanced dataset with 4 positives and 9 negatives. To improve the sampling speed, two lists of indices are maintained for the positive data and negative data, respectively. At the beginning, we shuffle the two lists and then take the first 4 positives and 4 negatives to form a mini-batch. Once the positive list is used up, we only reshuffle the positive list and take 4 shuffled positives to pair with next 4 negatives in the negative list as a mini-batch. Once the negative list is used up, we re-shuffle both lists and repeat the same process as above. An illustration of the impact of the DualSampler on the convergence is shown in Figure 6.15.

TriSampler. For multi-label classification problems with many labels and ranking problems, `TriSampler` first samples a set of tasks controlled by a hyperparameter `sampled_tasks`, and then sample positive and negative data for each task.

The following code snippet shows how to define `DualSampler` and `TriSampler`.

```
from libauc.sampler import DualSampler, TriSampler
dualsampler = DualSampler(trainSet,
                          batch_size=32,
                          sampling_rate=0.1)
trisampler = TriSampler(trainSet,
                        batch_size_per_task=32,
                        sampled_tasks=5,
                        sampling_rate_per_task=0.1)
```

Dynamic Mini-batch Loss

To compute the vanilla gradient estimator, we invoke backpropagation using the PyTorch function `loss.backward()` on a defined loss. The vanilla gradient estimators for pAUC, AP, and NDCG maximization share a common structure of the form

$$\frac{1}{|\mathcal{B}_1|}\sum_{\mathbf{x}_i\in\mathcal{B}_1}\frac{1}{|\mathcal{B}_2|}\sum_{\mathbf{x}_j\in\mathcal{B}_2} p_{ij}\nabla\ell(h(\mathbf{w};\mathbf{x}_j)-h(\mathbf{w};\mathbf{x}_i)),$$

where the weights p_{ij} are computed from dynamic variables within the algorithm. To enable the use of `loss.backward()`, it suffices to define a mini-batch loss as $\frac{1}{|\mathcal{B}_1|}\sum_{\mathbf{x}_i\in\mathcal{B}_1}\frac{1}{|\mathcal{B}_2|}\sum_{\mathbf{x}_j\in\mathcal{B}_2} p_{ij}\ell(h(\mathbf{w};\mathbf{x}_j)-h(\mathbf{w};\mathbf{x}_i))$, where p_{ij} is detached from the computation graph to avoid unnecessary backpropagation through these variables. Since p_{ij} is evolving across iterations, the mini-batch loss is called dynamic mini-batch loss. A high-level pseudocode example for SOPAs is provided in Figure 6.16.

```
# define dynamic mini-batch loss
def pAUCLoss(**kwargs):  # dynamic mini-batch loss
    sur_loss = surrogate_loss(neg_logits - pos_logits)
    exp_loss = torch.exp(sur_loss / Lambda)
    u[index] = (1 - gamma) * u[index] + gamma * (exp_loss.mean(1)
        )
    p = (exp_loss / u[index]).detach()
    loss = torch.mean(p * sur_loss)
    return loss

# optimization
for data, targets, index in dataloader:
    logits = model(data)
    loss = pAUCLoss(logits, targets, index)
    optimizer.zero_grad()
    loss.backward()
    optimizer.step()
```

Fig. 6.16: High-level pseudocode for SOPAs.

Comparison with Existing Libraries

We present some benchmark results of LibAUC in comparison with other state-of-the-art training libraries.

Comparison with the TFCO Library. We compare LibAUC (SOAP) with Google's TensorFlow Constrained Optimization (TFCO) library for optimizing average precision (AP). Both methods are trained for 100 epochs using a batch size of 128, the Adam optimizer with a learning rate of 1e-3, and a weight decay of 1e-4 on a binary classification task derived from CIFAR-10 with `imratio` $\in \{1\%, 2\%\}$. The training and testing learning curves, shown in Figure 6.11, demonstrate that LibAUC consistently outperforms TFCO.

Comparison with the TF-Ranking Library. We evaluate LibAUC, using SONG for NDCG maximization, against Google's TF-Ranking library, which implements `ApproxNDCG` and `GumbelNDCG`. Experiments are conducted on two large-scale datasets

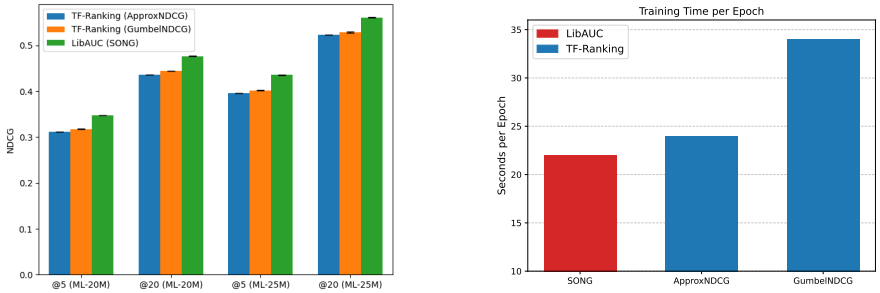


Fig. 6.17: Left: Benchmarks of NDCG optimization on MovieLens (ML) 20M and 25M datasets, @K means NDCG at top K. Right: Runtime Comparison between LibAUC and TF-ranking for NDCG maximization. For more details, please refer to (Yuan et al., 2023b).

—MovieLens20M and MovieLens25M—from the MovieLens platform. As shown in Figure 6.17, LibAUC achieves superior performance on both datasets. Furthermore, the runtime comparison shows that LibAUC's NDCG maximization algorithm is more efficient than the corresponding implementations in TF-Ranking.

6.5 Discriminative Pretraining of Representation Models

In Chapter 2, we briefly introduced the core concepts of representation learning and highlighted its growing significance in modern AI systems. In contemporary AI, representation models are learned through Self-supervised learning (SSL), which has emerged as a powerful paradigm for learning representation models without the need for labeled data. Among the most prominent frameworks within SSL is *contrastive learning*, which forms positive pairs by applying different augmentations to the same data sample or taking different views of the same data, while treating different data as negatives. In this section, we delve deeper into contrastive learning, with a focus on its applications to both unimodal and multimodal representation learning.

6.5.1 Mini-batch Contrastive Losses

A contrastive loss is used to pull the representations of positive pairs closer together, while pushing apart those of negative pairs in the embedding space. One of the most widely used contrastive losses is the so-called InfoNCE loss, which operates over samples within a mini-batch. Below, we illustrate its use in two well-known contrastive learning methods and discuss its limitations.

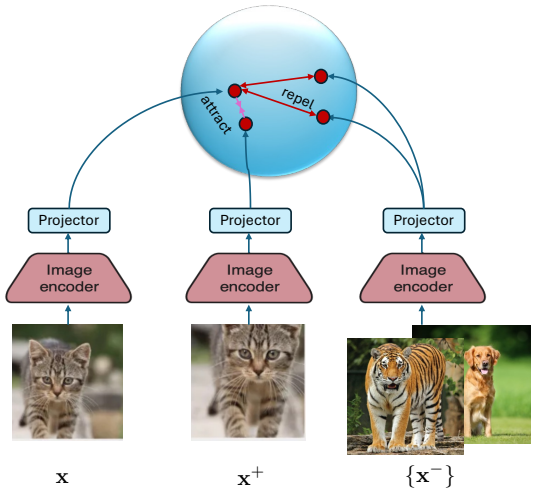


Fig. 6.18: Illustration of SimCLR for Contrastive Visual Representation Learning. $(\mathbf{x}, \mathbf{x}^+)$ are augmentations of the same image, $\{\mathbf{x}^-\}$ is a set of other images. An image encoder is a deep neural network and a projector is a lightweight multi-layer perceptron.

SimCLR

We now illustrate the contrastive loss in the context of visual representation learning by the well-known method SimCLR. The framework is illustrated in Figure 6.18. The model typically consists of a deep encoder backbone followed by a small projector, often implemented as a multi-layer perceptron (MLP). During downstream tasks, the projector is discarded, and the encoder's output is used as the final representation. The inclusion of the projector during training improves the quality and transferability of the learned embeddings by helping disentangle the contrastive learning objective from the representation space.

Let $(\mathbf{x}, \mathbf{x}^+) \sim \mathbb{P}_+$ denote a positive pair, which are different augmented copies from the same data. For a mini-batch $\mathcal{B} = \{\mathbf{x}_1, \ldots, \mathbf{x}_B\}$, each anchor $\mathbf{x}_i$ is paired with an augmented positive sample $\mathbf{x}_i^+$. The resulting mini-batch-based contrastive loss (commonly referred to as the InfoNCE loss) for anchor $\mathbf{x}_i$ is given by:

$$L_{\mathcal{B}}(\mathbf{w}; \mathbf{x}_i, \mathbf{x}_i^+) = -\log \frac{\exp\left(\frac{h(\mathbf{w};\mathbf{x}_i)^\top h(\mathbf{w};\mathbf{x}_i^+)}{\tau}\right)}{\exp\left(\frac{h(\mathbf{w};\mathbf{x}_i)^\top h(\mathbf{w};\mathbf{x}_i^+)}{\tau}\right) + \sum_{\mathbf{x}_j \in \mathcal{B}_i^-} \exp\left(\frac{h(\mathbf{w};\mathbf{x}_i)^\top h(\mathbf{w};\mathbf{x}_j)}{\tau}\right)}, \tag{6.38}$$

where $h(\mathbf{w}; \mathbf{x})$ denotes the normalized embedding of input $\mathbf{x}$, i.e., $\|h(\mathbf{w}; \mathbf{x})\|_2 = 1$, and $\tau > 0$ is the temperature parameter. The set $\mathcal{B}_i^-$ includes all negative samples in

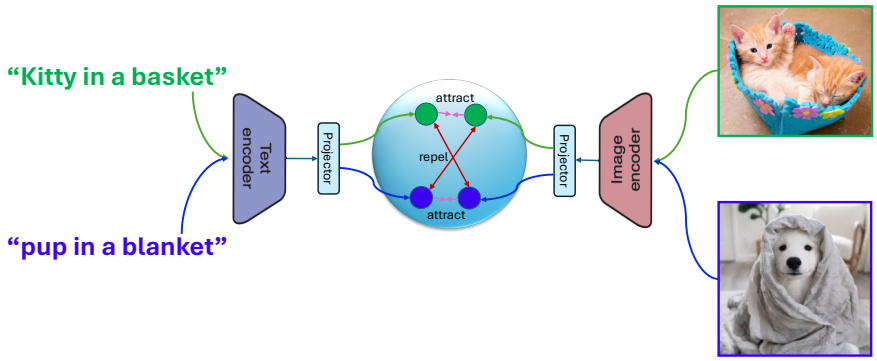


Fig. 6.19: Illustration of Contrastive Language-Image Pretraining (CLIP). A projector is usually a single linear layer.

the mini-batch excluding $\mathbf{x}_i$ and its augmentations. The positive pair can be removed from the denominator.

CLIP (Contrastive Language–Image Pretraining)

CLIP is a multimodal representation model that aligns images and text via contrastive learning on large-scale image–caption datasets. It comprises an image encoder and a text encoder, each followed by a corresponding projector, all jointly trained through contrastive learning (see Figure 6.19). CLIP models are typically trained on millions to billions of image–caption pairs, denoted as $\mathcal{S} = \{(\mathbf{x}_1, \mathbf{t}_1), \ldots, (\mathbf{x}_n, \mathbf{t}_n)\}$. Let $h_1(\mathbf{w}; \cdot)$ denote the image encoder and $h_2(\mathbf{w}; \cdot)$ denote the text encoder, which outputs normalized embedding vectors.

With a mini-batch $\mathcal{B} = \{(\mathbf{x}_1, \mathbf{t}_1), \ldots, (\mathbf{x}_B, \mathbf{t}_B)\}$, a mini-batch-based contrastive loss for each image $\mathbf{x}_i$ is given by:

$$L_{\mathcal{B}}(\mathbf{w}; \mathbf{x}_i) = -\log \frac{\exp\left(\frac{h_1(\mathbf{w};\mathbf{x}_i)^\top h_2(\mathbf{w};\mathbf{t}_i)}{\tau}\right)}{\exp\left(\frac{h_1(\mathbf{w};\mathbf{x}_i)^\top h_2(\mathbf{w};\mathbf{t}_i)}{\tau}\right) + \sum_{\mathbf{t}_j \in \mathcal{B}_{2i}^-} \exp\left(\frac{h_1(\mathbf{w};\mathbf{x}_i)^\top h_2(\mathbf{w};\mathbf{t}_j)}{\tau}\right)}, \quad (6.39)$$

where the set $\mathcal{B}_{2i}^-$ includes all negative texts in the mini-batch excluding $\mathbf{t}_i$. Similarly, a mini-batch-based contrastive loss for each caption $\mathbf{t}_i$ is given by:

$$L_{\mathcal{B}}(\mathbf{w}; \mathbf{t}_i) = -\log \frac{\exp\left(\frac{h_1(\mathbf{w};\mathbf{x}_i)^\top h_2(\mathbf{w};\mathbf{t}_i)}{\tau}\right)}{\exp\left(\frac{h_1(\mathbf{w};\mathbf{x}_i)^\top h_2(\mathbf{w};\mathbf{t}_i)}{\tau}\right) + \sum_{\mathbf{x}_j \in \mathcal{B}_{1i}^-} \exp\left(\frac{h_1(\mathbf{w};\mathbf{x}_j)^\top h_2(\mathbf{w};\mathbf{t}_i)}{\tau}\right)}. \quad (6.40)$$

where the set $\mathcal{B}_{1i}^-$ includes all negative images in the mini-batch excluding $\mathbf{x}_i$. Back-propagation is then performed on the two mini-batch contrastive losses to compute gradient estimators, which are summed to update the model parameters.

CLIP enables zero-shot image classification, cross-modality retrieval and plays a crucial role in text-to-image generation by guiding models to synthesize images that semantically align with textual prompts.

What is zero-shot classification?

Zero-shot classification means classifying data without any labeled data for learning a classifier. In a multi-class classification task with K classes $\{C_1, \ldots, C_K\}$, where each class corresponds to a specific label (e.g., 'dog'), we apply the CLIP model by first constructing a natural language prompt for each category (e.g., 'a photo of a dog'). We then compute text embeddings for these prompts and calculate their cosine similarity with the image embedding generated by CLIP. Finally, the model predicts the class that yields the highest similarity score.

The Challenge of Large Batch Size

While efficient, the InfoNCE loss is known to heavily rely on large batch sizes to ensure a rich and diverse set of negatives. For example, SimCLR requires a batch size of 8192 to achieve state-of-the-art performance for training on the ImageNet-1K dataset. This dependence on large batches imposes significant memory and computational burdens, especially when using large network backbones or processing high-dimensional inputs such as videos. Indeed, optimizing the InfoNCE loss is equivalent to using the BSGD method for optimizing the global contrastive loss as discussed in next subsection, which sufffers from non-convergence if the batch size is not significantly large.

6.5.2 Contrastive Learning without Large Batch Sizes

While the mini-batch contrastive loss offers computational convenience, it contradicts to the standard optimization principle where the objective is typically defined over the full dataset, followed by the development of efficient optimization algorithms. The mini-batch contrastive loss emerged naturally from the prevalent training pipeline (see Figure 6.1) that practitioners are familiar with. However, as previously discussed, this pipeline originating from ERM assumes that the loss for each data instance is independent of others, which does not hold for contrastive objectives. To resolve this, it is essential to decouple the design of the objective function from the optimization procedure.

Global Contrastive Loss: Separating Objective from Optimization

A global contrastive loss contrasts each anchor data point against all other examples in the training set. For a given positive pair $(\mathbf{x}_i, \mathbf{x}_i^+)$, the global contrastive loss is defined as:

$$L(\mathbf{w}; \mathbf{x}_i, \mathbf{x}_i^+) = \tau \log\left(\frac{1}{|\mathcal{S}_i^-|}\sum_{\mathbf{x}_j \in \mathcal{S}_i^-} \exp\left(\frac{h(\mathbf{w}; \mathbf{x}_i)^\top h(\mathbf{w}; \mathbf{x}_j) - h(\mathbf{w}; \mathbf{x}_i)^\top h(\mathbf{w}; \mathbf{x}_i^+)}{\tau}\right)\right), \tag{6.41}$$

where $\mathcal{S}_i^-$ is the set of all negative samples excluding $\mathbf{x}_i$ and its positive counterparts. The full global contrastive objective over $\mathcal{S} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ is then given by:

$$\min_{\mathbf{w}} F(\mathbf{w}) = \frac{1}{n}\sum_{\mathbf{x}_i \in \mathcal{S}} \frac{1}{|\mathcal{S}_i^+|}\sum_{\mathbf{x}_i^+ \in \mathcal{S}_i^+} L(\mathbf{w}; \mathbf{x}_i, \mathbf{x}_i^+), \tag{6.42}$$

where $\mathcal{S}_i^+$ denotes the set of all positive samples corresponding to $\mathbf{x}_i$.

SogCLR: The Optimization Algorithm

To optimize the global contrastive objective, we cast it into the following:

$$\begin{aligned}\min_{\mathbf{w}} & -\frac{1}{n}\sum_{\mathbf{x}_i \in \mathcal{S}} \frac{1}{|\mathcal{S}_i^+|}\sum_{\mathbf{x}_i^+ \in \mathcal{S}_i^+} h(\mathbf{w}; \mathbf{x}_i)^\top h(\mathbf{w}; \mathbf{x}_i^+) \\ & + \frac{1}{n}\sum_{\mathbf{x}_i \in \mathcal{S}} \log\left(\sum_{\mathbf{x}_j \in \mathcal{S}_i^-} \exp\left(\frac{h(\mathbf{w}; \mathbf{x}_i)^\top h(\mathbf{w}; \mathbf{x}_j)}{\tau}\right)\right).\end{aligned} \tag{6.43}$$

The first term is a standard average and the second term is an objective of FCCO, where the outer function is $f(\cdot) = \tau \log(\cdot)$ and the inner function is $g_i(\mathbf{w}) = \frac{1}{|\mathcal{S}_i^-|}\sum_{\mathbf{z} \in \mathcal{S}_i^-} \exp\left(\frac{h(\mathbf{w}; \mathbf{x}_i)^\top h(\mathbf{w}; \mathbf{z})}{\tau}\right)$. For readers who are familiar with Chapter 4 and 5, it is easy to understand the challenge of optimizing the above objective. It lies at the compositional structure of the second term with both summations over many data outside and inside the log function. As a result, the using the mini-batch-based InfoNCE loss will suffer from a biased gradient estimator whose error depends on the batch size.

To address this challenge, we can extend the SOX algorithm to solving (6.43) as shown in Algorithm 34, which is referred to as SogCLR. The estimators $u_{i,t+1}, \forall i$ are for tracking the inner function values $g_i(\mathbf{w}_t)$ and $p_{i,t} = \frac{1}{\varepsilon + u_{i,t+1}}$ is for estimating $\nabla \log(g_i(\mathbf{w}_t))$, where ε is small positive value added to avoid numerical issue and facilitate the learning.

Algorithm 34 SogCLR for optimizing the global contrastive objective (6.43)

Input: Initial model $\mathbf{w}_1$, $\mathbf{u}_0 \in \mathbb{R}^n$
for $t = 1$ to T **do**
  Sample a mini-batch $\mathcal{B} = \{\mathbf{x}_i\}_{i=1}^B$ with augmentations
  for each $\mathbf{x}_i \in \mathcal{B}$ **do**
    Construct the positive and negative set within mini-batch $\mathcal{B}_i^+$, $\mathcal{B}_i^-$
    Update $u_{i,t}$ via:

$$u_{i,t} = (1-\gamma)u_{i,t-1} + \gamma \frac{1}{|\mathcal{B}_i^-|} \sum_{\mathbf{z}\in\mathcal{B}_i^-} \exp\left(\frac{h(\mathbf{w}_t;\mathbf{x}_i)^\top h(\mathbf{w}_t;\mathbf{z})}{\tau}\right)$$

  end for
  Compute the vanilla gradient estimator $\mathbf{z}_t$:

$$\mathbf{z}_t = -\frac{1}{|\mathcal{B}|}\sum_{\mathbf{x}_i\in\mathcal{B}} \frac{1}{|\mathcal{B}_i^+|} \sum_{\mathbf{x}_i^+\in\mathcal{B}_i^+} \nabla(h(\mathbf{w}_t;\mathbf{x}_i)^\top h(\mathbf{w}_t;\mathbf{x}_i^+))$$
$$+ \frac{1}{|\mathcal{B}|}\sum_{\mathbf{x}_i\in\mathcal{B}} \frac{1}{|\mathcal{B}_i^-|} \sum_{\mathbf{z}\in\mathcal{B}_i^-} \frac{\exp\left(\frac{h(\mathbf{w}_t;\mathbf{x}_i)^\top h(\mathbf{w}_t;\mathbf{z})}{\tau}\right)}{\varepsilon + u_{i,t}} \nabla(h(\mathbf{w};\mathbf{x}_i)^\top h(\mathbf{w};\mathbf{z})),$$

  Update $\mathbf{w}_{t+1}$ by Momentum, Adam or AdamW
end for

Initialization and Update of u

Unlike the model parameter $\mathbf{w}$, which is typically initialized randomly, the auxiliary variables $\mathbf{u}$ can be initialized upon their first update. Specifically, when an index i is sampled for the first time, we set $\mathbf{u}_{i,t}$ to the corresponding mini-batch estimate of the inner function value.

As with the practical considerations discussed for distributionally robust optimization (DRO), the vanilla update of $\mathbf{u}$ can suffer from numerical instability due to the use of $\exp(\cdot)$, particularly when the temperature τ is small. To address this, we can instead maintain a log-transformed variable $\nu_{i,t} = \log u_{i,t}$, following the technique in Equation (6.14).

PyTorch Implementation

A PyTorch implementation of SogCLR for self-supervised visual representation learning is shown in Figure 6.21. Each image in the dataset is augmented twice. To facilitate the computation of the vanilla gradient estimator, we define a dynamic contrastive loss function. For each augmented instance, we call this loss function to update its associated u variable and compute the dynamic loss using the updated u. These individual dynamic losses are then aggregated over the mini-batch, and the u variables for the two augmentations of each image are averaged.

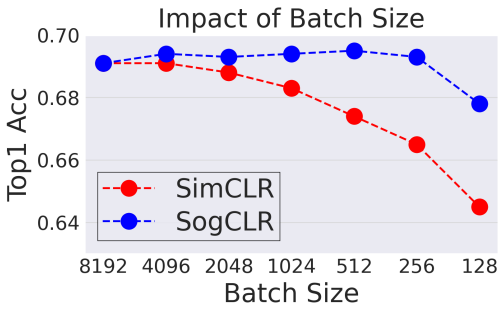


Fig. 6.20: Impact of batch size for different methods. The x-axis represents the batch size, and the y-axis shows the linear evaluation accuracy on the ImageNet validation set. Models were pretrained for 800 epochs using a ResNet-50 backbone on ImageNet-1K. For more details, please refer to (Yuan et al., 2022c).

Finally, we invoke `loss.backward()` to compute the gradient, followed by an optimizer step to update model parameters.

Comparison with SimCLR

The effectiveness of SogCLR is illustrated in Figure 6.20 with comparison with SimCLR for self-supervised visual representation learning on ImageNet-1K dataset with 1.2 million of images. With a standard mini-batch size 256 and the same other settings as SimCLR, by running 800 epochs, SogCLR achieves a performance of 69.4% for top 1 linear evaluation accuracy, which is better than 69.3% of SimCLR using a large batch size 8,192. Linear evaluation accuracy is measured by training a linear classifier atop a frozen encoder and subsequently assessing its performance on the validation set.

6.5.3 Contrastive Learning with Learnable Temperatures

The temperature parameter τ plays a critical role in controlling the penalty strength on negative samples. Specifically, a small τ penalizes much more on hard negative samples (i.e., the degree of hardness-awareness is high), causing separable embedding space. However, the excessive pursuit to the separability may break the underlying *semantic structures* because some negative samples with high similarity scores to the anchor data might indeed contain similar semantics, to which we refer as false negatives. In contrast, a large τ tends to treat all negative pairs equally (i.e., the degree of hardness-awareness is low) and is more tolerant to false negative samples, which is beneficial for keeping local semantic structures.

Existing approaches based on the InfoNCE loss often treat the temperature parameter τ as a learnable scalar to be optimized. However, this strategy lacks theoretical justification and may not yield optimal performance. Moreover, real-world data distributions typically exhibit long-tail characteristics, with substantial variation in the

```
# Note: This is a simplified version of SogCLR, we compute u
# from each augmentation separately for computing the dynamic
    contrastive loss
# and then  aggregated them from all augmentations.
# model: encoder + mlp projectors
# aug: a set of augmentation functions
# tau: temperature
# N: data size
# ind: indices for images in mini-batch
# u: 1d tensor with shape (N,1) by zero initialization
# g: parameter for maintaining moving averages of u

for ind, img in dataloader:
    x1, x2 = aug(img), aug(img)    # augmentations
    h1, h2 = model(x1), model(x2)  # forward pass
    h1, h2 = h1.norm(dim=1, p=2), h2.norm(dim=1, p=2)
    loss1, u1 = dcl(h1, h2, ind)   # dcl for h1, h2
    loss2, u2 = dcl(h2, h1, ind)   # dcl for h2, h1
    u[ind] = (u1 + u2)/2           # update u
    loss = (loss1 + loss2).mean()  # symmetrized
    loss.backward()
    update(model.params)           # momentum or adam-style

# dynamic contrastive loss (mini-batch)
def dcl(h1, h2, ind):
    B = h1.shape[0]
    labels = cat([one_hot(range(B)), one_hot(range(B))], dim=1)
    logits = cat([dot(h1, h2.T), dot(h1, h1.T)], dim=1)
    neg_logits = exp(logits/tau)*(1-labels)
    u_ = (1-g) * u[ind] + g*sum(neg_logits, dim=1)/(2(B-1))
    p = (neg_logits/u_).detach()
    sum_neg_logits = sum(p*logits, dim=1)/(2(B-1))
    normalized_logits = logits - sum_neg_logits
    loss = -sum(labels * normalized_logits, dim=1)
    return loss, u_
```

Fig. 6.21: PyTorch-style implementation of SogCLR for global contrastive learning.

frequency of samples across different semantic categories. This diversity suggests the need for individualized temperature parameters that better adapt to the inherent heterogeneity of the data.

To improve feature qualities, samples with frequent semantics should be assigned with a *large* τ to better capture the local semantic structure, while using a small τ will push semantically consistent samples away. On the other hand, samples with rare semantics should have a *small* τ to make their features more discriminative and separable.

Robust Global Contrastive Loss with a Learnable Temperature

Owing to the equivalence between the global contrastive loss and KL-regularized DRO (see Eq. (2.14)), the loss in Eq. (6.41) can be rewritten as:

$$L(\mathbf{w}; \mathbf{x}_i, \mathbf{x}_i^+) = \max_{\mathbf{p} \in \Delta} \sum_{\mathbf{x}_j \in \mathcal{S}_i^-} p_j \left(h(\mathbf{w}; \mathbf{x}_i)^\top h(\mathbf{w}; \mathbf{x}_j) - h(\mathbf{w}; \mathbf{x}_i)^\top h(\mathbf{w}; \mathbf{x}_i^+) \right) - \tau \, \mathrm{KL}(\mathbf{p}, 1/|\mathcal{S}_i^-|), \tag{6.44}$$

where Δ is the probability simplex over $\mathcal{S}_i^-$ and τ serves as the regularization parameter in the KL-regularized DRO.

To enable learning of the temperature parameter, we formulate a robust global contrastive loss using a KL-constrained DRO framework:

$$\begin{aligned} &\hat{L}(\mathbf{w}; \mathbf{x}_i, \mathbf{x}_i^+) = \\ &\max_{\mathbf{p} \in \Delta} \sum_{\mathbf{x}_j \in \mathcal{S}_i^-} p_j \left(h(\mathbf{w}; \mathbf{x}_i)^\top h(\mathbf{w}; \mathbf{x}_j) - h(\mathbf{w}; \mathbf{x}_i)^\top h(\mathbf{w}; \mathbf{x}_i^+) \right) - \tau_0 \, \mathrm{KL}(\mathbf{p}, 1/|\mathcal{S}_i^-|) \\ &\text{subject to} \quad \mathrm{KL}(\mathbf{p}, 1/|\mathcal{S}_i^-|) \le \rho, \end{aligned} \tag{6.45}$$

where τ_0 is a small constant to ensure smoothness of $\hat{L}(\mathbf{w}; \mathbf{x}_i, \mathbf{x}_i^+)$. Using the dual formulation (cf. Eq. (2.19)), this can be equivalently expressed as:

$$\hat{L}(\mathbf{w}; \mathbf{x}_i, \mathbf{x}_i^+) = \min_{\tau \ge \tau_0} \tau \log \left(\frac{1}{|\mathcal{S}_i^-|} \sum_{\mathbf{x}_j \in \mathcal{S}_i^-} \exp \left(\frac{h(\mathbf{w}; \mathbf{x}_i)^\top h(\mathbf{w}; \mathbf{x}_j) - h(\mathbf{w}; \mathbf{x}_i)^\top h(\mathbf{w}; \mathbf{x}_i^+)}{\tau} \right) \right) + \tau \rho. \tag{6.46}$$

Let $\ell_i(\mathbf{w}; \mathbf{x}_j) = h(\mathbf{w}; \mathbf{x}_i)^\top h(\mathbf{w}; \mathbf{x}_j) - h(\mathbf{w}; \mathbf{x}_i)^\top h(\mathbf{w}; \mathbf{x}_i^+)$. The above loss simplifies further to:

$$\hat{L}(\mathbf{w}; \mathbf{x}_i, \mathbf{x}_i^+) = \min_{\tau \ge \tau_0} \tau \log \left(\frac{1}{|\mathcal{S}_i^-|} \sum_{\mathbf{x}_j \in \mathcal{S}_i^-} \exp \left(\frac{\ell_i(\mathbf{w}; \mathbf{x}_j)}{\tau} \right) \right) + \tau \rho.$$

Minimizing the average of these robust global contrastive losses yields the following objective, which learns individualized temperatures:

$$\min_{\mathbf{w}} \frac{1}{n} \sum_{\mathbf{x}_i \in \mathcal{S}} \left\{ \min_{\tau_i \ge \tau_0} \tau_i \log \left(\frac{1}{|\mathcal{S}_i^-|} \sum_{\mathbf{x}_j \in \mathcal{S}_i^-} \exp \left(\frac{\ell_i(\mathbf{w}; \mathbf{x}_j)}{\tau_i} \right) \right) + \tau_i \rho \right\}. \tag{6.47}$$

The SogCLR algorithm can be modified to solve this problem. We present the resulting algorithm, referred to as iSogCLR, in Algorithm 35. The vanilla gradient estimator with respect to $\mathbf{w}_t$ is computed as in SogCLR, except that the temperature τ is replaced with the individualized $\tau_{i,t}$ at iteration t. The gradient estimator with

Algorithm 35 iSogCLR for optimizing the robust global contrastive objective (6.47)

1: **Input:** Initial model $\mathbf{w}_1$, $\mathbf{u}_0 \in \mathbb{R}^n$
2: **for** $t = 1$ to T **do**
3: Sample a mini-batch $\mathcal{B} = \{\mathbf{x}_i\}_{i=1}^B$ with augmentations
4: **for** each $\mathbf{x}_i \in \mathcal{B}$ **do**
5: Construct the positive and negative set within mini-batch B_i^+, B_i^-
6: Update $u_{i,t}$ via:

$$u_{i,t} = (1-\gamma)u_{i,t-1} + \gamma \frac{1}{|\mathcal{B}_i^-|} \sum_{\mathbf{z}\in\mathcal{B}_i^-} \exp\left(\frac{\ell_i(\mathbf{w};\mathbf{z})}{\tau_{i,t}}\right)$$

7: Compute the vanilla gradient estimator $\mathbf{z}_{i,t}$ of $\tau_{i,t}$

$$\mathbf{z}_{i,t} = -\frac{1}{|\mathcal{B}_i^-|} \sum_{\mathbf{z}\in\mathcal{B}_i^-} \frac{\exp\left(\frac{\ell_i(\mathbf{w};\mathbf{z})}{\tau_{i,t}}\right)}{\varepsilon + u_{i,t}} \frac{\ell_i(\mathbf{w};\mathbf{z})}{\tau_{i,t}} + \log(u_{i,t}) + \rho$$

8: **end for**
9: Compute the vanilla gradient estimators $\mathbf{z}_t$:

$$\mathbf{z}_t = \frac{1}{|\mathcal{B}|} \sum_{\mathbf{x}_i\in\mathcal{B}} \frac{1}{|\mathcal{B}_i^-|} \sum_{\mathbf{z}\in\mathcal{B}_i^-} \frac{\exp\left(\frac{\ell_i(\mathbf{w}_t;\mathbf{z})}{\tau}\right)}{\varepsilon + u_{i,t}} \nabla \ell_i(\mathbf{w}_t;\mathbf{z}),$$

10: Update $\tau_{i,t+1}$, $\forall \mathbf{x}_i \in \mathcal{B}$ by the Momentum method
11: Update $\mathbf{w}_{t+1}$ by the Momentum or AdamW method
12: **end for**

respect to $\tau_{i,t}$ is computed in Step 7 and it can be updated using the Momentum method.

An application of iSogCLR to CIFAR-10 dataset yields more discriminative features than SimCLR and SogCLR as shown in Figure 6.22.

CLIP Training with Learnable Temperatures

CLIP with Individualized Learnable Temperatures

We can integrate the robust global contrastive loss for temperature learning into the contrastive language-image pretraining (CLIP), yielding the following objective:

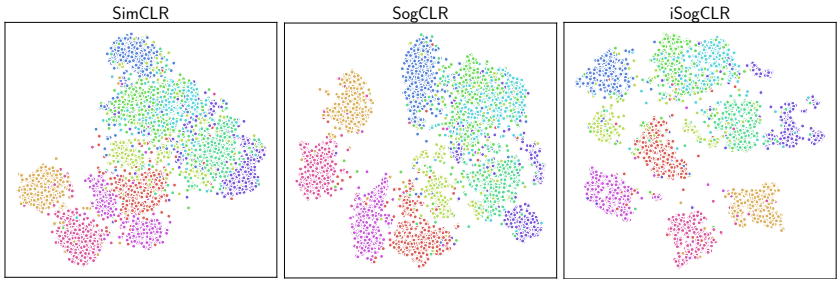


Fig. 6.22: The learned embeddings (projected onto 2D space using t-SNE) for CIFAR10 samples learned by self-supervised learning algorithms SimCLR, SogCLR and iSogCLR. For more details, please refer to (Qiu et al., 2023).

$$\min_{\mathbf{w},\boldsymbol{\tau}_1\geq\tau_0,\boldsymbol{\tau}_2\geq\tau_0} \frac{1}{n}\sum_{i=1}^{n}\tau_{i,1}\log\left(\frac{1}{|\mathcal{T}_i^-|}\sum_{\mathbf{t}\in\mathcal{T}_i^-}\exp\left(\frac{s(\mathbf{w};\mathbf{x}_i,\mathbf{t})-s(\mathbf{w};\mathbf{x}_i,\mathbf{t}_i)}{\tau_{i,1}}\right)\right)+\tau_{i,1}\rho$$
$$+\frac{1}{n}\sum_{i=1}^{n}\tau_{i,2}\log\left(\frac{1}{|\mathcal{I}_i^-|}\sum_{\mathbf{x}\in\mathcal{I}_i^-}\exp\left(\frac{s(\mathbf{w};\mathbf{x},\mathbf{t}_i)-s(\mathbf{w};\mathbf{x}_i,\mathbf{t}_i))}{\tau_{i,2}}\right)\right)+\tau_{i,2}\rho, \tag{6.48}$$

where $\mathcal{T}_i^-$ denotes the set of all negative data of an image $\mathbf{x}_i$ and $\mathcal{I}_i^-$ denotes the set of all negative data of the corresponding text $\mathbf{t}_i$, and $s(\mathbf{w};\mathbf{x},\mathbf{t}) = h_1(\mathbf{w};\mathbf{x})^\top h_2(\mathbf{w};\mathbf{t})$ is the similarity score of the image and text embeddings.

While optimizing robust contrastive losses enables the learning of temperature parameters, it may compromise generalizability in downstream tasks by introducing a large number of additional parameters, which can lead to overfitting—particularly in noisy real-world datasets where mismatched samples are common. Two approaches can be used to tackle this issue.

CLIP with a Global Learnable Temperature

A straightforward approach to reduce the number of temperature parameters is to learn a single global temperature parameter for images and texts, respectively. This is formulated as the following optimization problem:

$$\min_{\mathbf{w},\tau_1\geq\tau_0,\tau_2\geq\tau_0} \frac{1}{n}\sum_{i=1}^{n}\left\{\tau_1\log\left(\frac{1}{|\mathcal{T}_i^-|}\sum_{\mathbf{t}\in\mathcal{T}_i^-}\exp\left(\frac{s(\mathbf{w};\mathbf{x}_i,\mathbf{t})-s(\mathbf{w};\mathbf{x}_i,\mathbf{t}_i)}{\tau_1}\right)\right)+\tau_1\rho\right\}$$
$$+\frac{1}{n}\sum_{i=1}^{n}\left\{\tau_2\log\left(\frac{1}{|\mathcal{I}_i^-|}\sum_{\mathbf{x}\in\mathcal{I}_i^-}\exp\left(\frac{s(\mathbf{w};\mathbf{x},\mathbf{t}_i)-s(\mathbf{w};\mathbf{x}_i,\mathbf{t}_i)}{\tau_2}\right)\right)+\tau_2\rho\right\}. \tag{6.49}$$

CLIP with a Temperature Prediction Network

An alternative strategy is to learn a temperature prediction network (TempNet) that outputs an instance-dependent temperature for each image and text. The corresponding optimization problem is defined as:

$$\min_{\mathbf{w},\mathbf{w}_1',\mathbf{w}_2'} \frac{1}{n}\sum_{i=1}^{n} \tau(\mathbf{w}_1';\mathbf{x}_i)\log\left(\frac{1}{|\mathcal{T}_i^-|}\sum_{\mathbf{t}\in\mathcal{T}_i^-}\exp\left(\frac{s(\mathbf{w};\mathbf{x}_i,\mathbf{t})-s(\mathbf{w};\mathbf{x}_i,\mathbf{t}_i)}{\tau(\mathbf{w}_1';\mathbf{x}_i)}\right)\right)+\tau(\mathbf{w}_1';\mathbf{x}_i)\rho$$
$$+\frac{1}{n}\sum_{i=1}^{n} \tau(\mathbf{w}_2';\mathbf{t}_i)\log\left(\frac{1}{|\mathcal{I}_i^-|}\sum_{\mathbf{x}\in\mathcal{I}_i^-}\exp\left(\frac{s(\mathbf{w};\mathbf{x},\mathbf{t}_i)-s(\mathbf{w};\mathbf{x}_i,\mathbf{t}_i)}{\tau(\mathbf{w}_2';\mathbf{t}_i)}\right)\right)+\tau(\mathbf{w}_2';\mathbf{t}_i)\rho. \tag{6.50}$$

The temperature prediction network $\tau(\mathbf{w}_1';\cdot)$ for images can share the encoder layers of the image encoder $h_1(\mathbf{w};\cdot)$, followed by a lightweight MLP. Similarly, the text-side temperature prediction network $\tau(\mathbf{w}_2';\cdot)$ can share the encoder layers of the text encoder $h_2(\mathbf{w};\cdot)$, also followed by a small MLP. Again this problem can be optimized by modifying SogCLR to account for the update of TempNet.

Scheduler of γ

Like the standard learning rate η in the update of $\mathbf{w}_{t+1}$, the hyper-parameter γ can be also interpreted as a learning rate of SGD (4.3). The theoretical analysis shows that γ should be set to a very small value close to 0 in order to guarantee convergence. Ideally, γ should be large to rely more on the current mini-batch at earlier iterations and be smaller to rely more on history in later iterations. To achieve this, we can use a decreasing scheduler, e.g., a cosine schedule for γ_t: Let t be the current iteration, t_0 be the number of iterations per epoch and E be the number of decay epochs, then we set $\gamma_t = 0.5\cdot(1+\cos(\pi\lfloor t/t_0\rfloor/E))\cdot(1-\gamma_{\min})+\gamma_{\min}$. With this schedule, γ_t will decrease from 1.0 to $\gamma_{\min}$. Note that $\lfloor t/t_0\rfloor$ denotes the current epoch, which means the value of γ_t stays unchanged within one epoch. Also, The number of decay epochs E is a hyperparameter, and it is not necessarily equal to the total number of training epochs. If the current epoch exceeds E, γ_t will be set to $\gamma_{\min}$.

PyTorch Implementations

PyTorch implementations of SogCLR and iSogCLR are available in the LibAUC library. Their distributed versions, including support for solving (6.49) with a cosine scheduler for γ, are provided in the FastCLIP GitHub repository:

https://github.com/Optimization-AI/FastCLIP

Three versions are available: FastCLIP-v1 implements SogCLR with a tuned global temperature, FastCLIP-v2 implements iSogCLR with individualized temperatures,

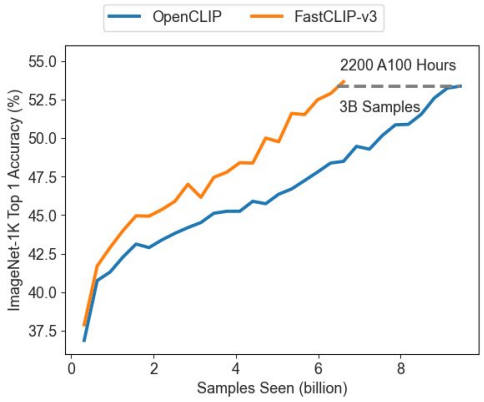


Fig. 6.23: FastCLIP-v3 vs OpenCLIP. The training was conducted on LAION315M with 315M image-text pairs for learning ViT-B/16 using a total of 5120 batch size on 8 A100. Y-axis is the zero-shot accuracy on ImageNet validation data. For more details, please refer to (Wei et al., 2024).

and FastCLIP-v3 implements SogCLR for solving the global temperature optimization in (6.49).

A distributed implementation of iSogCLR for CLIP training with the Temperature Prediction Network (TempNet) is available at:

`https://github.com/Optimization-AI/DistTempNet`

Figure 6.23 presents a comparison between FastCLIP-v3 and the prior state-of-the-art distributed implementation of optimizing the mini-batch-based InfoNCE loss, known as OpenCLIP (Ilharco et al., 2021). This highlights the effectiveness of the advanced compositional optimization algorithm, demonstrating clear improvements in both convergence speed and representation quality.

6.6 Discriminative Fine-tuning of Large Language Models

Large Language Models (LLMs) have revolutionized modern AI. Their training typically consists of three stages: self-supervised pretraining on internet-scale text corpora, supervised fine-tuning (SFT) on question–answer datasets, and learning with human preference for alignment. An improved paradigm, *reinforcement learning with verifiable rewards* (RLVR), further advances large reasoning models by leveraging automatically verifiable signals from synthesized outputs.

6.6.1 Pipeline of LLM Training

Figure 6.24 illustrates the pipeline of LLM Training. We briefly introduce these components below.

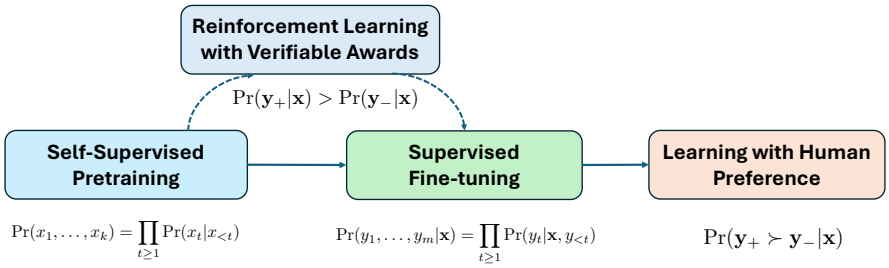


Fig. 6.24: Different Phases of training LLMs.

Self-supervised Pretraining

Self-supervised pretraining is formulated as next-token prediction. Let $\mathbf{x} = (x_1, \ldots, x_m)$ be a sequence of tokens where x_j belongs to a vocabulary of tokens $\mathcal{V} = \{v_1, \ldots, v_K\}$. The probability of $\mathbf{x}$ is modeled auto-regressively by

$$p(\mathbf{x}) = \prod_{j=1}^{m} p(x_j|x_{<j}),$$

where $x_{<j}$ denotes the prefix $(x_1, \ldots, x_{j-1})$. The conditional probability is modeled via a softmax over a Transformer representation:

$$p(x_j|x_{<j}) = \pi_{\mathbf{w}}(x_j|x_{<j}) = \frac{\exp(h(\mathbf{w}_0; x_{<j})^\top \mathbf{w}_{x_j})}{\sum_{k=1}^{K} \exp(h(\mathbf{w}_0; x_{<j})^\top \mathbf{w}_k)}, \tag{6.51}$$

where $h(\mathbf{w}_0; x_{<j}) \in \mathbb{R}^d$ is produced by a Transformer network and $\mathbf{w}_{x_j} \in \mathbb{R}^d$ is the token embedding. The full model parameters $\mathbf{w} = (\mathbf{w}_0, \mathbf{w}_1, \ldots, \mathbf{w}_K)$ are learned by minimizing the negative log-likelihood over a dataset $\mathcal{S} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$:

$$\min_{\mathbf{w}} -\frac{1}{n} \sum_{i=1}^{n} \log p(\mathbf{x}_i). \tag{6.52}$$

Supervised Fine-tuning (SFT)

In SFT, a dataset $\mathcal{S} = \{(\mathbf{x}_i, \mathbf{y}_i)\}$ is used, where $\mathbf{x}_i$ is an input prompt and $\mathbf{y}_i$ is the desired output. Let $\mathbf{x} = (x_1, \ldots, x_k)$ and $\mathbf{y} = (y_1, \ldots, y_{m'})$ be token sequences from the vocabulary $\mathcal{V}$. SFT models the next-token prediction of tokens in $\mathbf{y}$ given $\mathbf{x}$ using the autoregressive factorization:

$$p(\mathbf{y}|\mathbf{x}) = \prod_{j=1}^{m'} \pi_{\mathbf{w}}(y_j|\mathbf{x}, y_{<j}),$$

where each term is computed using the same Transformer-based model as in pre-training. SFT minimizes:

$$\min_{\mathbf{w}} -\frac{1}{n}\sum_{i=1}^{n} \log p(\mathbf{y}_i|\mathbf{x}_i). \tag{6.53}$$

Learning with Human Preference

SFT does not penalize poor responses. Hence, it does not necessarily guarantee that the likelihood of tokens in a poor answer is low. Let us consider a simple example:

Motivation Example
($\mathbf{x}$) What is the bigger number between 9.11 and 9.9? ($\mathbf{y}$) The bigger number between 9.11 and 9.9 is 9.9. ($\mathbf{y}'$) The bigger number between 9.11 and 9.9 is 9.11.

The good answer $\mathbf{y}$ and the bad answer $\mathbf{y}'$ only differ in the last token. The likelihood of all preceding tokens are the same. Even though the likelihood of the last token "9" in $\mathbf{y}$ conditioned on preceding tokens is increased during the fine-tuning with this data, the likelihood of the token "11" as the last one might still be high, making generating the bad answer $\mathbf{y}'$ likely.

To address this issue, learning with human feedback fine-tunes the model using preference tuples $(\mathbf{x}, \mathbf{y}_+, \mathbf{y}_-)$, where $\mathbf{y}_+$ is preferred over $\mathbf{y}_-$. Two main approaches are reinforcement learning from human feedback (RLHF) and direct preference optimization (DPO).

RLHF

A reward model $r_\theta(\mathbf{x}, \mathbf{y})$ is first trained to match human preferences by modeling the preference probability $\Pr(\mathbf{y}_+ \succ \mathbf{y}_-|\mathbf{x})$ as

$$p(\mathbf{y}_+ \succ \mathbf{y}_-|\mathbf{x}) = \frac{\exp(r_\theta(\mathbf{x}, \mathbf{y}_+))}{\exp(r_\theta(\mathbf{x}, \mathbf{y}_+)) + \exp(r_\theta(\mathbf{x}, \mathbf{y}_-))}, \tag{6.54}$$

and minimizing the following:

$$\min_{\theta} \mathbb{E}_{\mathbf{x},\mathbf{y}_+,\mathbf{y}_-} - \log p(\mathbf{y}_+ \succ \mathbf{y}_-|\mathbf{x}). \tag{6.55}$$

The policy model (i.e, the target LLM) is then optimized by solving the following problem with some RL algorithms:

$$\max_{\mathbf{w}} \mathbb{E}_{\mathbf{x},\mathbf{y}\sim\pi_{\mathbf{w}}} \left[r_{\theta_*}(\mathbf{x}, \mathbf{y}) - \beta \mathrm{KL}(\pi_{\mathbf{w}}(\cdot|\mathbf{x}), \pi_{\mathrm{ref}}(\cdot|\mathbf{x})) \right]. \tag{6.56}$$

where the KL divergence is defined as:

$$\mathrm{KL}(\pi_{\mathbf{w}}(\cdot|\mathbf{x}), \pi_{\mathrm{ref}}(\cdot|\mathbf{x})) = \mathbb{E}_{\mathbf{y}\sim\pi_{\mathbf{w}}(\cdot|\mathbf{x})}\left[\log \frac{\pi_{\mathbf{w}}(\mathbf{y}|\mathbf{x})}{\pi_{\mathrm{ref}}(\mathbf{y}|\mathbf{x})}\right], \tag{6.57}$$

where π_{ref} denotes a base model. If we decompose $\mathbf{y} = (y_1, \ldots, y_k)$ as a sequence of tokens, then using the autoregressive factorization the KL divergence can be expressed as a sum over tokens:

$$\mathrm{KL}(\pi_{\mathbf{w}}(\cdot|\mathbf{x}), \pi_{\mathrm{ref}}(\cdot|\mathbf{x})) = \mathbb{E}_{\mathbf{y}\sim\pi_{\mathbf{w}}}\left[\sum_{t=1}^{k} \log \frac{\pi_{\mathbf{w}}(y_t|\mathbf{x}, y_{<t})}{\pi_{\mathrm{ref}}(y_t|\mathbf{x}, y_{<t})}\right]. \tag{6.58}$$

Direct Preference Optimization (DPO)

DPO directly optimizes the policy without a separate reward model. A closed-form non-parameterized solution of π by solving (6.56) for any reward model $r(\mathbf{x}, y)$, gives:

$$\pi(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})}\pi_{\mathrm{ref}}(\mathbf{y}|\mathbf{x}) \exp(\beta r(\mathbf{x}, \mathbf{y})), \tag{6.59}$$

where $Z(\mathbf{x})$ is the normalization factor. Substituting into Eq. (6.55) leads to:

$$\min_{\mathbf{w}} \mathbb{E}_{\mathbf{x},\mathbf{y}_+,\mathbf{y}_-} \log\left(1 + \exp\left(\beta \log \frac{\pi_{\mathbf{w}}(\mathbf{y}_-|\mathbf{x})}{\pi_{\mathrm{ref}}(\mathbf{y}_-|\mathbf{x})} - \beta \log \frac{\pi_{\mathbf{w}}(\mathbf{y}_+|\mathbf{x})}{\pi_{\mathrm{ref}}(\mathbf{y}_+|\mathbf{x})}\right)\right). \tag{6.60}$$

In practice, a set of tuples $\{(\mathbf{x}_i, \mathbf{y}_{i+}, \mathbf{y}_{i-}\}_{i=1}^n$ is constructed and used for learning.

Connections with Discriminative Learning and AUC Maximization

DPO can be also motivated from discriminative learning, particularly AUC maximization. We view generating the answers of $\mathbf{x}$ as a task, and $\mathbf{y}_+$ denotes a positive data and $\mathbf{y}_-$ denotes a negative data. Let $s(\mathbf{w}, \mathbf{x}, \mathbf{y})$ denote a scoring function, which indicates the likelihood of generating $\mathbf{y}$ given $\mathbf{x}$. By AUC maximization with a continuous surrogate loss $\ell(s(\mathbf{w}, \mathbf{x}, \mathbf{y}_-) - s(\mathbf{w}, \mathbf{x}, \mathbf{y}_+))$, we have the following problem:

$$\min_{\theta} \mathbb{E}_{\mathbf{x},\mathbf{y}_+,\mathbf{y}_-} \ell(s(\mathbf{w}, \mathbf{x}, \mathbf{y}_-) - s(\mathbf{w}, \mathbf{x}, \mathbf{y}_+)). \tag{6.61}$$

DPO can be recovered by setting $s(\mathbf{w}, \mathbf{x}, \mathbf{y}) = \log \frac{\pi(\mathbf{y}|\mathbf{x})}{\pi_{\mathrm{ref}}(\mathbf{y}|\mathbf{x})}$ and $\ell(s) = \log(1 + \exp(\beta s))$.

Reinforcement Learning with Verifiable Rewards (RLVR)

RLVR is an emerging paradigm for training reasoning models, particularly suited for tasks like mathematical problem solving, where models are expected to generate step-by-step solutions followed by a final answer. Unlike RLHF, which relies on

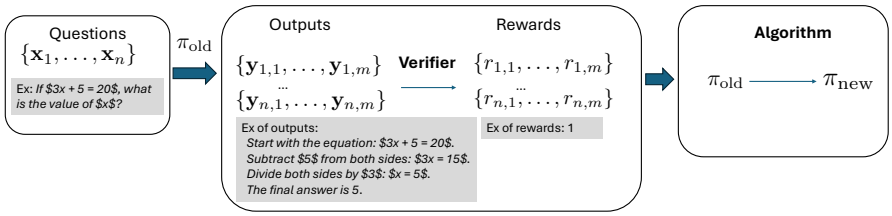


Fig. 6.25: The one-step iteration of RL for reinforcing Large Reasoning Model. For each question $\mathbf{x}_i$, the model generates m outputs $\mathbf{y}_{i,1}, \ldots, \mathbf{y}_{i,m}$ and each of them receives a reward $r_{i,j}$, $j = 1, \ldots, m$ from a verifier. Then an algorithm will leverage the inputs, their outputs and the reward information to update the model.

subjective preference labels, RLVR leverages verifiable signals such as whether the final answer is correct.

What is a Large Reasoning Model?

A large reasoning model is a type of LLM that is specifically designed or fine-tuned to perform multi-step logical reasoning, such as solving math problems, answering complex questions, or generating structured arguments. It generates intermediate reasoning tokens before producing the final answer, mimicking System 2 reasoning in humans, which is deliberate, logical, and slow.

RLVR is illustrated in Figure 6.25. The old model in one step of learning is denoted by π_{old}. It is used to generate multiple answers for a set of input questions. Given a question $\mathbf{x}$ (with prompt included), one generated output $\mathbf{y}$ follows the distribution $\pi_{\text{old}}(\cdot|\mathbf{x})$, which includes reasoning traces and the final answer. Specifically, output $\mathbf{y}$ is generated token by token, i.e., $y_t \sim \pi_{\text{old}}(\cdot|\mathbf{x}, y_{<t})$, for $t = 1, \cdots, |\mathbf{y}|$.

A key to RLVR is to assume that there exists a verifier, which can automatically verifies the quality of the generated answer, giving a reward. Let us consider a binary reward setting where the verifier returns a binary value for a given question $\mathbf{x}$ and its corresponding answer in the output $\mathbf{y}$. For answering mathematical questions, this can be achieved by comparing the generated answer with the true answer. For generating mathematical proofs, we can use a formal verification tool such as LEAN to verify if the proof is correct.

Proximal Policy Optimization (PPO)

PPO is a classical RL algorithm. Let

$$\rho_{\mathbf{w}}(\mathbf{x}, \mathbf{y}) = \frac{\pi_{\mathbf{w}}(\mathbf{y}|\mathbf{x})}{\pi_{\text{old}}(\mathbf{y}|\mathbf{x})}$$

denote the likelihood ratio between the new policy $\pi_{\mathbf{w}}$ and the old policy π_{old}. Let $A(\mathbf{x}, \mathbf{y})$ be an advantage function for taking action $\mathbf{y}$ given input $\mathbf{x}$, which measures how much better a specific action is compared to the policy's average behavior in a given state. The PPO objective is given by:

$$\begin{aligned}\mathcal{L}_{\text{PPO}}(\mathbf{w}) =& \mathbb{E}_{\mathbf{x},\mathbf{y}\sim\pi_{\text{old}}}\left[\min\left(\rho_{\mathbf{w}}(\mathbf{x},\mathbf{y})\cdot A(\mathbf{x},\mathbf{y}),\ \text{clip}(\rho_{\mathbf{w}}(\mathbf{x},\mathbf{y}), 1-\epsilon, 1+\epsilon)\cdot A(\mathbf{x},\mathbf{y})\right)\right], \\ & - \beta \text{KL}(\pi_{\mathbf{w}}, \pi_{\text{ref}}),\end{aligned} \tag{6.62}$$

where $\epsilon > 0$ is a small hyperparameter (typically around 0.1 or 0.2), and the `clip` function restricts the likelihood ratio $\rho_{\mathbf{w}}(\mathbf{x}, \mathbf{y})$ to the range $[1-\epsilon, 1+\epsilon]$, defined as:

$$\text{clip}(\rho_{\mathbf{w}}(\mathbf{x},\mathbf{y}), 1-\epsilon, 1+\epsilon) = \begin{cases} 1-\epsilon & \text{if } \rho_{\mathbf{w}}(\mathbf{x},\mathbf{y}) < 1-\epsilon, \\ \rho_{\mathbf{w}}(\mathbf{x},\mathbf{y}) & \text{if } 1-\epsilon \le \rho_{\mathbf{w}}(\mathbf{x},\mathbf{y}) \le 1+\epsilon, \\ 1+\epsilon & \text{if } \rho_{\mathbf{w}}(\mathbf{x},\mathbf{y}) > 1+\epsilon. \end{cases}$$

The intuition of using clipping mechanism is that

- When $A(\mathbf{x}, \mathbf{y}) > 0$ (the action is better than expected), the clip operation prevents $\pi_{\mathbf{w}}$ from increasing its probability too aggressively.
- When $A(\mathbf{x}, \mathbf{y}) < 0$ (the action is worse than expected), the clip operation prevents $\pi_{\mathbf{w}}$ from decreasing its probability too drastically.

This clipping mechanism was used to reduce variance and maintain stable training dynamics for reinforcement learning. However, it also suffers from zero gradient when $\rho_{\mathbf{w}}(\mathbf{x}, \mathbf{y})$ is out of the range $[1-\epsilon, 1+\epsilon]$, which might slow down the learning process.

Trust Region Policy Optimization (TRPO)

TRPO is a principled policy optimization method that improves stability and efficiency by restricting each policy update to stay within a small trust region. It maximizes a surrogate objective function based on the advantage estimates under the old policy, while constraining the average Kullback–Leibler (KL) divergence between the old and new policies. Formally, TRPO solves the following constrained optimization problem:

$$\begin{aligned} \max_{\theta} \quad & \mathbb{E}_{\mathbf{x},\mathbf{y}\sim\pi_{\text{old}}}\left[\rho_{\mathbf{w}}(\mathbf{x},\mathbf{y}) A(\mathbf{x},\mathbf{y})\right] \\ \text{subject to} \quad & \mathbb{E}_{\mathbf{x}}\left[\text{KL}\left(\pi_{\text{old}}(\cdot|\mathbf{x}), \pi_{\mathbf{w}}(\cdot|\mathbf{x})\right)\right] \le \delta, \end{aligned} \tag{6.63}$$

where δ is a predefined trust region threshold. The KL divergence is taken in the reverse direction to ensure that the updated policy does not deviate too much from the old policy on average across the state distribution.

Group Relative Policy Optimization (GRPO).

GRPO is a reinforcement learning algorithm designed to optimize policies by leveraging group-wise relative reward information.

For inputs $\{\mathbf{x}_i\}_{i=1}^m$, let $\{\mathbf{y}_{ij}\}_{j=1}^K$ denote the corresponding set of K generated answers for each $\mathbf{x}_i$. the objective of GRPO for maximization is defined by:

$$\begin{aligned}\mathcal{J}_{\text{GRPO}}(\mathbf{w}) =& \frac{1}{m}\sum_{i=1}^{m}\frac{1}{k}\sum_{j=1}^{k}\left[\frac{1}{|\mathbf{y}_{ij}|}\sum_{t=1}^{|\mathbf{y}_{ij}|} f\left(\frac{\pi_{\mathbf{w}}(y_{ij,t}|\mathbf{x}, y_{ij,<t})}{\pi_{\text{old}}(y_{ij,t}|\mathbf{x}, y_{ij,<t})}, A(\mathbf{x}_i, \mathbf{y}_{ij})\right)\right] \\ & - \beta \text{KL}(\pi_\theta, \pi_{\text{ref}}),\end{aligned} \tag{6.64}$$

where $y_{ij,t}$ denotes its t-th token and $y_{ij,<t}$ denotes the prefix of the t-th token of $\mathbf{y}_{ij}$, $f(s,t) = \min(st, \text{clip}(s, 1-\epsilon, 1+\epsilon)t)$, π_{ref} is a frozen reference model, and $A(\mathbf{x}_i, \mathbf{y}_{ij})$ is the group-wise advantage function defined as

$$A(\mathbf{x}, \mathbf{y}) = \frac{r(\mathbf{y}|\mathbf{x}) - \bar{r}_q}{\sigma_q}$$

with $\bar{r}_q$ being the average reward of outputs for $\mathbf{x}$ and σ_q being its standard deviation. This advantage function quantifies how much better the reward of an output $\mathbf{y}$ is compared to average reward in the group. For analysis, we consider the expected version:

$$\mathcal{J}_{\text{GRPO}}(\mathbf{w}) = \mathbb{E}_{\mathbf{x}}\mathbb{E}_{\mathbf{y}\sim\pi_{\text{old}}(\cdot|\mathbf{x})}\left[\frac{1}{|\mathbf{y}|}\sum_{t=1}^{|\mathbf{y}|} f\left(\frac{\pi_{\mathbf{w}}(y_t|\mathbf{x}, y_{<t})}{\pi_{\text{old}}(y_t|\mathbf{x}, y_{<t})}, A(\mathbf{x}, \mathbf{y})\right)\right] - \beta \text{KL}(\pi_\theta, \pi_{\text{ref}}), \tag{6.65}$$

where

$$A(\mathbf{x}, \mathbf{y}) = \frac{r(\mathbf{y}|\mathbf{x}) - \mathbb{E}_{\mathbf{y}'\sim\pi_{\text{old}}(\cdot|\mathbf{x})} r(\mathbf{y}'|\mathbf{x})}{\sqrt{\text{Var}_{\mathbf{y}'\sim\pi_{\text{old}}(\cdot|\mathbf{x})} r(\mathbf{y}'|\mathbf{x})}}. \tag{6.66}$$

6.6.2 DFT for fine-tuning Large Language Models

While learning with human feedback addresses the limitation of SFT, traditional supervised learning methods never use human preference data. For example, in image classification, training data $(\mathbf{x}, y)$ denote an input image and its true class label $y \in \{1, \ldots, K\}$. We do not need the preference optimization step on preference data saying that a dog class is preferred to a cat class for an image of a dog. So what is the difference between traditional supervised learning and supervised finetuning of LLMs that makes SFT not enough? The answer lies in the fact that traditional supervised learning methods are usually **discriminative approaches**, while the SFT method is not discriminative.

By casting the supervised fine-tuning of LLMs into data prediction, we can leverage discriminative learning approaches, e.g., the discriminative probabilistic modeling (DPM) approach and the robust optimization approach.

DPM over an Infinite Data Space

Let $\mathcal{X}$ and $\mathcal{Y}$ be infinite data spaces. Let us consider $\mathcal{X}$ as an anchor space and $\mathcal{Y}$ as the target space with a Lebesgue measure μ. When $\mathcal{Y}$ is countably infinite, the Lebesgue measure μ is replaced by the counting measure. We model the probability density $\Pr(\mathbf{y} \mid \mathbf{x})$ of an object $\mathbf{y} \in \mathcal{Y}$ given an anchor object $\mathbf{x} \in \mathcal{X}$ by a parameterized scoring function $s(\mathbf{w}; \mathbf{x}, \mathbf{y})$:

$$P_{\mathbf{w}}(\mathbf{y} \mid \mathbf{x}) = \frac{\exp(s(\mathbf{w}; \mathbf{x}, \mathbf{y})/\tau)}{\int_{\mathcal{Y}} \exp(s(\mathbf{w}; \mathbf{x}, \mathbf{y}')/\tau) d\mu(\mathbf{y}')}, \tag{6.67}$$

where $\tau > 0$ is a temperature parameter. We assume that $\exp(s(\mathbf{w}; \mathbf{x}, \mathbf{y})/\tau)$ is Lebesgue-integrable for $\mathbf{w} \in \mathcal{W}$, $\mathcal{W} \subset \mathbb{R}^d$. Here $P_{\mathbf{w}}(\mathbf{y} \mid \mathbf{x})$ is a valid probability density function because $\int_{\mathcal{Y}} P_{\mathbf{w}}(\mathbf{y} \mid \mathbf{x}) d\mu(\mathbf{y}) = 1$. Given $\{(\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_n, \mathbf{y}_n)\}$ sampled from the joint distribution $p_{\mathbf{x},\mathbf{y}}$, the maximum likelihood estimation (MLE) can be formulated as the following:

$$\begin{aligned} &\min_{\mathbf{w}} \left\{ -\frac{1}{n} \sum_{i=1}^{n} \tau \log \frac{\exp(s(\mathbf{w}; \mathbf{x}, \mathbf{y})/\tau)}{\int_{\mathcal{Y}} \exp(s(\mathbf{w}; \mathbf{x}, \mathbf{y}')/\tau) d\mu(\mathbf{y}')} \right\} \\ &= -\frac{1}{n} \sum_{i=1}^{n} s(\mathbf{w}; \mathbf{x}, \mathbf{y}) + \tau \log \left(\int_{\mathcal{Y}} \exp(s(\mathbf{w}; \mathbf{x}, \mathbf{y}')/\tau) d\mu(\mathbf{y}') \right). \end{aligned} \tag{6.68}$$

If $\mathcal{Y}$ is finite, the above DPM framework recovers the traditional multi-class classification and learning to rank. In particular, if $\mathcal{Y}$ denotes the label set $\{1, \ldots, K\}$ and $s(\mathbf{w}; \mathbf{x}, y)$ denotes the classification score for the y-th class, then the above approach recovers logistic regression. If $\mathcal{Y}$ denotes the set of items $\mathcal{Y} = \{\mathbf{x}_{q,1}, \ldots, \mathbf{x}_{q,N_q}\}$ and the anchor data $\mathbf{x}$ denotes a query, then the above approach recovers the ListNet (2.47).

Optimization via FCCO

The main challenge for solving the DPM problem over an infinite data space lies in computing the integral $g(\mathbf{w}; \mathbf{x}_i, \mathcal{Y}) \coloneqq \int_{\mathcal{Y}} \exp\left(s(\mathbf{w}; \mathbf{x}_i, \mathbf{y}')/\tau\right) d\mu(\mathbf{y}')$ for each $i \in [n]$, which is infeasible unless $\mathcal{Y}$ is finite. Below, we discuss two general approaches for tackling the challenge.

Sample and Optimize

The first approach is to introduce a sampling distribution $P_i(\cdot)$, satisfying that (1) it is easy to sample data from P_i; (2) it is possible to compute the probability value of a sample $\mathbf{y}'$. Then we write

$$\int_{\mathcal{Y}} \exp\left(\frac{s(\mathbf{w};\mathbf{x}_i,\mathbf{y}')}{\tau}\right) d\mu(\mathbf{y}') = \mathbb{E}_{\mathbf{y}'\sim P_i(\cdot)} \frac{\exp(s(\mathbf{w};\mathbf{x}_i,\mathbf{y}')/\tau)}{P_i(\mathbf{y}')}.$$

The optimization problem becomes an instance of FCCO:

$$\begin{aligned}\min_{\mathbf{w}} & -\frac{1}{n}\sum_{i=1}^n s(\mathbf{w};\mathbf{y}_i,\mathbf{x}_i) \\ & + \frac{1}{n}\sum_{i=1}^n \tau \log\left(\mathbb{E}_{\mathbf{y}'\sim P_i(\cdot)} \frac{\exp(s(\mathbf{w};\mathbf{y}',\mathbf{x}_i)/\tau)}{P_i(\mathbf{y}')}\right).\end{aligned} \tag{6.69}$$

Approximate and Optimize

In some cases, we may only have sampled data from $P_i(\cdot)$ without access to $P_i(\cdot)$. Let $\mathcal{S}_i = \{\mathbf{y}'_{i,1},\ldots,\mathbf{y}'_{i,m}\}$ denote a set of outputs sampled for each data $\mathbf{x}_i$ following some P_i. Then we approximate $g(\mathbf{w};\mathbf{x}_i,\mathcal{Y})$ by

$$g(\mathbf{w};\mathbf{x}_i,\mathcal{Y}) \approx \frac{1}{m}\sum_{\mathbf{y}'\in\mathcal{S}_i^-} \frac{\exp(s(\mathbf{w};\mathbf{y}',\mathbf{x})/\tau)}{P_i(\mathbf{y}')} \propto \frac{1}{m}\sum_{\mathbf{y}'\in\mathcal{S}_i} \exp(\frac{s(\mathbf{w};\mathbf{y}',\mathbf{x})}{\tau}), \tag{6.70}$$

where the last step assumes $P_i(\mathbf{y}')$ are approximately equal. Then the optimization problem becomes an instance of FCCO:

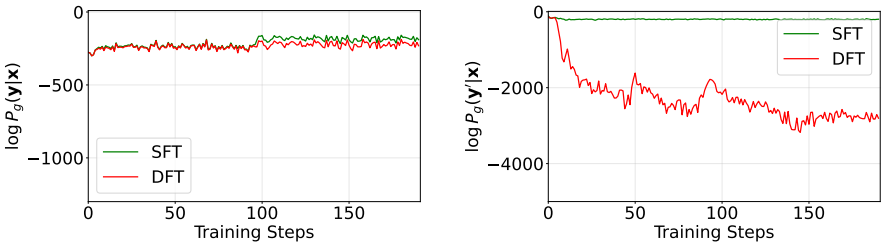
$$\begin{aligned}\min_{\theta} & -\frac{1}{n}\sum_{i=1}^n s(\mathbf{w};\mathbf{y}_i,\mathbf{x}_i) \\ & + \frac{1}{n}\sum_{i=1}^n \tau \log\left(\frac{1}{m}\sum\nolimits_{\mathbf{y}'\in\mathcal{S}_i} \exp(s(\mathbf{w};\mathbf{y}',\mathbf{x}_i)/\tau)\right).\end{aligned} \tag{6.71}$$

DFT for fine-tuning LLMs

Let us apply the DPM approach to fine-tuning LLMs, which is referred to as discriminative fine-tuning (DFT).

Discriminative Likelihood

Unlike SFT that maximizes the generative likelihood of tokens, DFT will maximize the discriminative likelihood of data as defined in (6.67). By maximizing the dis-



(a) Log-likelihoods of positives

(b) Log-likelihoods of negatives

Fig. 6.26: (a) Log-likelihoods of (annotated) positive examples during training for different methods. (b) Log-likelihoods of "negative" examples (generated from the base model) during training for different methods. For more details, please refer to (Guo et al., 2025).

Algorithm 36 The DFT Algorithm

Initialize $\mathbf{w}_1$ as the base LLM, and $\mathbf{u}_0 = \mathbf{1}$
for $t = 1, \ldots, T-1$ **do**
  Sample a mini-batch $\mathcal{B}_t \subset \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$
  for each $\mathbf{x}_i \in \mathcal{B}_t$ **do**
    Sample a mini-batch $\mathcal{B}_{i,t}^-$ from $\pi_{\text{ref}}(\cdot|\bar{\mathbf{x}}_i)$ via an offline pool
    Update $u_{i,t+1}$ according to

$$u_{i,t} = (1-\gamma)u_{i,t-1} + \gamma \frac{1}{B} \sum_{\mathbf{y}' \in \mathcal{B}_{i,t}^0} \frac{\exp(\frac{s(\mathbf{w}_t;\mathbf{y}',\mathbf{x}_i)}{\tau})}{\pi_{\text{ref}}(\mathbf{y}'|\bar{\mathbf{x}}_i)}, \tag{6.72}$$

  end for
  Compute a vanilla gradient estimator $\mathbf{z}_t$ according to

$$\mathbf{z}_t = -\frac{1}{|\mathcal{B}_t|} \sum_{\mathbf{x}_i \in \mathcal{B}_t} \nabla s(\mathbf{w}_t; \mathbf{y}_i, \mathbf{x}_i) + \frac{1}{|\mathcal{B}_t|} \sum_{\mathbf{x}_i \in \mathcal{B}_t} \frac{1}{u_{i,t+1}|\mathcal{B}_{i,t}^-|} \sum_{\mathbf{y}' \in \mathcal{B}_{i,t}^-} \frac{\exp(\frac{s(\mathbf{w}_t;\mathbf{y}',\mathbf{x}_i)}{\tau}) \nabla s(\mathbf{w}_t; \mathbf{y}', \mathbf{x}_i)}{\pi_{\text{ref}}(\mathbf{y}'|\bar{\mathbf{x}}_i)}. \tag{6.73}$$

  Update $\mathbf{w}_{t+1}$ using Momentum or AdamW
end for

criminative log-likelihood of the training data, we not only increase the score of the true output $\mathbf{y}_i$ for each input $\mathbf{x}_i$, corresponding to the numerator of the discriminative likelihood, but also decrease the scores of other potentially bad answers in $\mathcal{Y}$, which correspond to the denominator of the discriminative likelihood; see Figure 6.26.

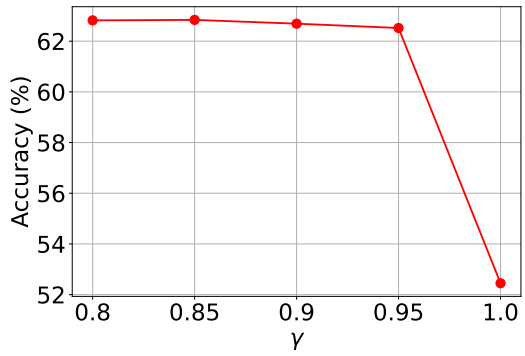


Fig. 6.27: Using moving average estimators with $\gamma < 1$ is important for improving the performance. For more details, please refer to (Guo et al., 2025).

The Scoring Function

For fine-tuning LLMs, the scoring function can be defined based on the generative log-likelihood $\log \pi_{\mathbf{w}}(\mathbf{y}|\mathbf{x})$, as it measures the likeliness of generating $\mathbf{y}$ given $\mathbf{x}$ by the model $\pi_{\mathbf{w}}$. For a good model, we expect that a high value of the generative log-likelihood $\log \pi_{\mathbf{w}}(\mathbf{y}|\mathbf{x})$ would indicate a high fitness score of $\mathbf{y}$ to answer $\mathbf{x}$. With such correspondence, the above discriminative learning framework would increase the chance of generating a good output $\mathbf{y}$ given $\mathbf{x}$ and decrease the chance of generating possibly bad outputs given $\mathbf{x}$. Common choices for the scoring function include the raw log-likelihood $s(\mathbf{w}; \mathbf{y}, \mathbf{x}) = \log \pi_{\mathbf{w}}(\mathbf{y}|\mathbf{x})$ and a length-normalized version $s(\mathbf{w}; \mathbf{y}, \mathbf{x}) = \frac{1}{|\mathbf{y}|} \log \pi_{\mathbf{w}}(\mathbf{y}|\mathbf{x})$. Using the unnormalized version $s_{\mathbf{w}}(\mathbf{y}, \mathbf{x}) = \log \pi_{\mathbf{w}}(\mathbf{y}|\mathbf{x})$ leads to the following DFT objective:

$$\begin{aligned} &\min_{\mathbf{w}} -\frac{1}{n} \sum_{i=1}^{n} \log \pi_{\mathbf{w}}(\mathbf{y}_i|\mathbf{x}_i) \\ &+ \tau \frac{1}{n} \sum_{i=1}^{n} \log \left(\sum\nolimits_{\mathbf{y}' \in \mathcal{Y}} \exp \left(\frac{\log \pi_{\mathbf{w}}(\mathbf{y}'|\mathbf{x}_i)}{\tau} \right) \right). \end{aligned} \tag{6.74}$$

Comparing the DFT objective of to that of SFT in (6.53), we observe that the first term in (6.74) is identical to the objective of SFT. The key difference lies in the second term, which penalizes the possibly poor outputs in $\mathcal{Y}$ for each $\mathbf{x}_i$ by reducing their generative log-likelihood, thereby discouraging their generation.

Sampling Distribution

The optimization analysis reveals that the variance bound σ_0 of the mini-batch estimator for the inner function $g(\mathbf{w}; \mathbf{x}_i, \mathcal{Y})$ significantly impacts convergence speed (cf. Theorem 5.1). Ideally, the variance-minimizing distribution is $P_{\mathbf{w}}(\cdot|\mathbf{x}_i)$. How-

ever, this distribution is impractical to evaluate and difficult to sample from directly. Moreover, we aim for the sampled outputs $\mathbf{y}' \sim P_i(\cdot)$ to represent likely poor responses to $\mathbf{x}_i$. A practical approach is to define $P_i(\cdot) = \pi_{\text{ref}}(\cdot|\bar{\mathbf{x}}_i)$, where π_{ref} denotes the base LLM to be fine-tuned and $\bar{\mathbf{x}}_i$ is an augmented version of $\mathbf{x}_i$ with added system prompts to encourage the generation of suboptimal outputs. This relies on the assumption that the base model is unlikely to generate high-quality answers in this context.

The Optimization Algorithm

An application of the SOX algorithm for solving (6.69) is presented in Algorithm 36. The sequence $\{u\}$ plays a critical role in effectively penalizing the sampled "negative data," as illustrated in Figure 6.27. A PyTorch implementation of DFT is at

https://github.com/Optimization-AI/DFT.

6.6.3 DisCO for Reinforcing Large Reasoning Models

DisCO, short for *Discriminative Constrained Optimization*, is a recent approach for reinforcing large reasoning models. It is motivated by the connection between the GRPO objective and discriminative learning objectives, and is designed to overcome key limitations of GRPO and its variants.

Limitation of GRPO and Connection with Discriminative Learning

Let $r(\mathbf{y}|\mathbf{x}) \in \{1, 0\}$ denote the reward assigned to an output $\mathbf{y}$ with respect to the input $\mathbf{x}$. A quantity that is important to the analysis is $p(\mathbf{x}) = \mathbb{E}_{\mathbf{y}\sim\pi_{\text{old}}(\cdot|\mathbf{x})}[r(\mathbf{y}|\mathbf{x})] \in [0, 1]$, which quantifies the difficulty of the question $\mathbf{x}$ under the model π_{old}. We denote by $\pi^+_{\text{old}}(\cdot|\mathbf{x})$ the conditional distribution of outputs when the reward is one (i.e., positive answers) and by $\pi^-_{\text{old}}(\cdot|\mathbf{x})$ the conditional distribution of outputs when the reward is zero (i.e., negative answers).

In the following analysis we assume $p(\mathbf{x}) = \mathbb{E}_{\mathbf{y}\sim\pi_{\text{old}}(\cdot|\mathbf{x})} r(\mathbf{y}|\mathbf{x}) \in (0, 1)$; otherwise we can remove them from consideration as done in practice.

Proposition 6.1. *Let us consider the objective of GRPO and its variants with the following form:*

$$\mathcal{J}_0(\mathbf{w}) = \mathbb{E}_{\mathbf{x}}\mathbb{E}_{\mathbf{y}\sim\pi_{old}(\cdot|\mathbf{x})}\left[\frac{1}{|\mathbf{y}|}\sum_{t=1}^{|\mathbf{y}|} f\left(\frac{\pi_{\mathbf{w}}(y_t|\mathbf{x}, y_{<t})}{\pi_{old}(y_t|\mathbf{x}, y_{<t})}, A(\mathbf{x}, \mathbf{y})\right)\right], \tag{6.75}$$

where $A(\mathbf{x}, \mathbf{y})$ is given in (6.66). Assume that $f(x, y)$ is non-decreasing function of x such that $f(x, y) = \mathbb{I}(y > 0)yf^+(x, 1) - \mathbb{I}(y \le 0)yf^-(x, 1)$, where both f^+, f^- are non-decreasing functions of x, then we have

$$\mathcal{J}_0(\mathbf{w}) = \mathbb{E}_{\mathbf{x}}\sqrt{p(\mathbf{x})(1-p(\mathbf{x}))}\mathbb{E}_{\mathbf{y}\sim\pi^+_{old}(\cdot|\mathbf{x}),\mathbf{y}'\sim\pi^-_{old}(\cdot|\mathbf{x})}[s^+(\mathbf{w};\mathbf{y},\mathbf{x}) - s^-(\mathbf{w};\mathbf{y}',\mathbf{x})], \tag{6.76}$$

where

$$s^+(\mathbf{w};\mathbf{y},\mathbf{x}) = \frac{1}{|\mathbf{y}|}\sum_{t=1}^{|\mathbf{y}|} f^+\left(\frac{\pi_{\mathbf{w}}(y_t|\mathbf{x},y_{<t})}{\pi_{old}(y_t|\mathbf{x},y_{<t})},1\right)$$
$$s^-(\mathbf{w};\mathbf{y},\mathbf{x}) = \frac{1}{|\mathbf{y}|}\sum_{t=1}^{|\mathbf{y}|} f^-\left(\frac{\pi_{\mathbf{w}}(y_t|\mathbf{x},y_{<t})}{\pi_{old}(y_t|\mathbf{x},y_{<t})},1\right).$$

In particular, for GRPO we have

$$s^+(\mathbf{w};\mathbf{y},\mathbf{x}) = \frac{1}{|\mathbf{y}|}\sum_{t=1}^{|\mathbf{y}|} \min(\frac{\pi_{\mathbf{w}}(y_t|\mathbf{x},y_{<t})}{\pi_{old}(y_t|\mathbf{x},y_{<t})},1+\epsilon), \tag{6.77}$$

$$s^-(\mathbf{w};\mathbf{y},\mathbf{x}) = \frac{1}{|\mathbf{y}|}\sum_{t=1}^{|\mathbf{y}|} \max(\frac{\pi_{\mathbf{w}}(y_t|\mathbf{x},y_{<t})}{\pi_{old}(y_t|\mathbf{x},y_{<t})},1-\epsilon). \tag{6.78}$$

Proof. Since $\mathbb{E}_{\mathbf{y}\sim\pi_{\text{old}}(\cdot|\mathbf{x})}r(\mathbf{y}|\mathbf{x}) = p(\mathbf{x}), \text{Var}_{\mathbf{y}\sim\pi_{\text{old}}(\cdot|\mathbf{x})}r(\mathbf{y}|\mathbf{x}) = p(\mathbf{x})(1-p(\mathbf{x}))$, we have

$$A(\mathbf{x},\mathbf{y}) = \begin{cases} \sqrt{\frac{1-p(\mathbf{x})}{p(\mathbf{x})}}, & \text{if } r(\mathbf{y}|\mathbf{x}) = 1, \\ -\sqrt{\frac{p(\mathbf{x})}{1-p(\mathbf{x})}}, & \text{if } r(\mathbf{y}|\mathbf{x}) = 0. \end{cases} \tag{6.79}$$

By the law of total expectation, we have

$$
\begin{aligned}
&\mathbb{E}_{\mathbf{x}}\mathbb{E}_{\mathbf{y}\sim\pi_{\text{old}}(\cdot|\mathbf{x})}\left[\frac{1}{|\mathbf{y}|}\sum_{t=1}^{|\mathbf{y}|} f\left(\frac{\pi_{\mathbf{w}}(y_t|\mathbf{x},y_{<t})}{\pi_{\text{old}}(y_t|\mathbf{x},y_{<t})}, A(\mathbf{x},\mathbf{y})\right)\right] \\
&= \mathbb{E}_{\mathbf{x}}\Bigg[p(\mathbf{x})\mathbb{E}_{\mathbf{y}\sim\pi^{+}_{\text{old}}(\cdot|\mathbf{x})}\frac{1}{|\mathbf{y}|}\sum_{t=1}^{|\mathbf{y}|} f\left(\frac{\pi_{\mathbf{w}}(y_t|\mathbf{x},y_{<t})}{\pi_{\text{old}}(y_t|\mathbf{x},y_{<t})}, A(\mathbf{x},\mathbf{y})\right) \\
&\qquad + (1-p(\mathbf{x}))\mathbb{E}_{\mathbf{y}\sim\pi^{-}_{\text{old}}(\cdot|\mathbf{x})}\frac{1}{|\mathbf{y}|}\sum_{t=1}^{|\mathbf{y}|} f\left(\frac{\pi_{\mathbf{w}}(y_t|\mathbf{x},y_{<t})}{\pi_{\text{old}}(y_t|\mathbf{x},y_{<t})}, A(\mathbf{x},\mathbf{y})\right)\Bigg] \\
&= \mathbb{E}_{\mathbf{x}}\Bigg[p(\mathbf{x})\mathbb{E}_{\mathbf{y}\sim\pi^{+}_{\text{old}}(\cdot|\mathbf{x})}\frac{1}{|\mathbf{y}|}\sum_{t=1}^{|\mathbf{y}|} f\left(\frac{\pi_{\mathbf{w}}(y_t|\mathbf{x},y_{<t})}{\pi_{\text{old}}(y_t|\mathbf{x},y_{<t})}, \sqrt{\frac{1-p(\mathbf{x})}{p(\mathbf{x})}}\right) \\
&\qquad + (1-p(\mathbf{x}))\mathbb{E}_{\mathbf{y}\sim\pi^{-}_{\text{old}}(\cdot|\mathbf{x})}\frac{1}{|\mathbf{y}|}\sum_{t=1}^{|\mathbf{y}|} f\left(\frac{\pi_{\mathbf{w}}(y_t|\mathbf{x},y_{<t})}{\pi_{\text{old}}(y_t|\mathbf{x},y_{<t})}, -\sqrt{\frac{p(\mathbf{x})}{1-p(\mathbf{x})}}\right)\Bigg] \\
&= \mathbb{E}_{\mathbf{x}}\sqrt{p(\mathbf{x})(1-p(\mathbf{x}))}\Bigg[\mathbb{E}_{\mathbf{y}\sim\pi^{+}_{\text{old}}(\cdot|\mathbf{x})}\frac{1}{|\mathbf{y}|}\sum_{t=1}^{|\mathbf{y}|} f^{+}\left(\frac{\pi_{\mathbf{w}}(y_t|\mathbf{x},y_{<t})}{\pi_{\text{old}}(y_t|\mathbf{x},y_{<t})}, 1\right) \\
&\qquad - \mathbb{E}_{\mathbf{y}\sim\pi^{-}_{\text{old}}(\cdot|\mathbf{x})}\frac{1}{|\mathbf{y}|}\sum_{t=1}^{|\mathbf{y}|} f^{-}\left(\frac{\pi_{\mathbf{w}}(y_t|\mathbf{x},y_{<t})}{\pi_{\text{old}}(y_t|\mathbf{x},y_{<t})}, 1\right)\Bigg],
\end{aligned}
\tag{6.80}
$$

where the last equality follows from the assumption about $f(x, y)$. For GPRO, we have $f^{+}(x, 1) = \min(x, \text{clip}(x, 1-\epsilon, 1+\epsilon)) = \min(x, 1+\epsilon)$ and $f^{-}(x, 1) = \max(x, \text{clip}(x, 1-\epsilon, 1+\epsilon)) = \max(x, 1-\epsilon)$. □

Why it matters

We derive two insights from Proposition 6.1 regarding the two components of $\mathcal{J}_0$. First, let us consider the component $\mathbb{E}_{\mathbf{y}\sim\pi^{+}_{\text{old}}(\cdot|\mathbf{x}),\mathbf{y}'\sim\pi^{-}_{\text{old}}(\cdot|\mathbf{x}),}[s^{+}(\mathbf{w};\mathbf{y},\mathbf{x}) - s^{-}(\mathbf{w};\mathbf{y}',\mathbf{x})]$. Since both f^{+} and f^{-} are non-decreasing functions of the first argument, then both $s^{+}(\mathbf{w};\mathbf{y},\mathbf{x})$ and $s^{-}(\mathbf{w};\mathbf{y},\mathbf{x})$ are non-decreasing functions of $\pi_\theta(y_t|\mathbf{x},y_{<t})$. Hence, maximizing $\mathcal{J}_0$ would increase the likelihood of tokens in the positive answers and decrease the likelihood of tokens in the negative answers. This makes sense as we would like the new model to have a high likelihood of generating a positive (correct) answer and a low likelihood of generating a negative (incorrect) answer. This mechanism is closely related to traditional discriminative methods of supervised learning in the context of AUC maximization, which aims to maximize the scores of positive samples $\mathbf{y}\sim\pi^{+}_{\text{old}}(\cdot|\mathbf{x})$ while minimizing scores of negative samples $\mathbf{y}'\sim\pi^{-}_{\text{old}}(\cdot|\mathbf{x})$, where the $\mathbf{x}$ acts like the classification task in the AUC maximization. Hence, in the context of discriminative learning, we refer to $s^{+}(\mathbf{y},\mathbf{x})$ and $s^{-}(\mathbf{y},\mathbf{x})$ as scoring functions. Therefore, $\mathbb{E}_{\mathbf{y}\sim\pi^{+}_{\text{old}}(\cdot|\mathbf{x}),\mathbf{y}'\sim\pi^{-}_{\text{old}}(\cdot|\mathbf{x}),}[s^{+}(\mathbf{y},\mathbf{x}) - s^{-}(\mathbf{y}',\mathbf{x})]$ is a discriminative objective.

Second, let us consider the component $\omega(\mathbf{x}) = \sqrt{p(\mathbf{x})(1-p(\mathbf{x}))}$, which acts like a weight scaling the discriminative objective for each individual input question.

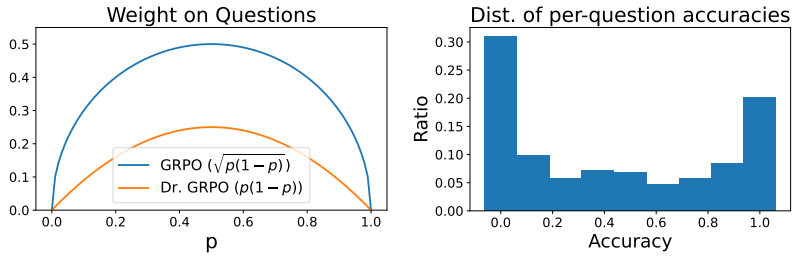


Fig. 6.28: (a) Weight on questions based on correctness probability p; (b) Histogram of per-question accuracy evaluated in the GRPO learning.

It is this component that leads to difficulty bias. As shown in Figure 6.28(a), questions with very high $p(\mathbf{x})$ values (close to 1) or very low $p(\mathbf{x})$ values (close to 0) receive small weights for their discriminative objectives, causing the optimization to focus primarily on questions of intermediate difficulty while paying little attention to hard questions ($p(\mathbf{x}) \approx 0$) and easy questions ($p(\mathbf{x}) \approx 1$). This mechanism may significantly hinder the learning efficiency. Intuitively, if the generated answers have only one correct solution out of 10 trials, i.e. $p(\mathbf{x}) = 0.1$, we should grasp this chance to enhance the model instead of overlooking it. On the other hand, even when we encounter an easy question with a probability of $p(\mathbf{x}) = 0.9$, we should keep improving the model rather than being satisfied because it still makes mistakes with respect to this question.

DisCO: A Discriminative Constrained Optimization Framework

Motivated by the analysis of GRPO and its connection with discriminative learning, discriminative objectives can be borrowed directly for learning the reasoning model. Below, we introduce two approaches.

Discriminative Objectives

For a given question $\mathbf{x}$, let $s(\mathbf{w}; \mathbf{y}, \mathbf{x})$ denote a scoring function that measures how likely the model $\pi_{\mathbf{w}}$ "predicts" the output $\mathbf{y}$ for a given input $\mathbf{x}$ [1]. Then the AUC score for the "task" $\mathbf{x}$ is equivalent to $\mathbb{E}_{\mathbf{y}\sim\pi^+_{\text{old}},\mathbf{y}'\sim\pi^-_{\text{old}}}[\mathbb{I}(s(\mathbf{w};\mathbf{y},\mathbf{x}) > s(\mathbf{w};\mathbf{y}'\mathbf{x}))]$. Using a non-decreasing continuous surrogate function ℓ, we form the following objective (in expectation form) for minimization:

$$\mathcal{L}_1(\mathbf{w}) := \mathbb{E}_{\mathbf{x}}\mathbb{E}_{\mathbf{y}\sim\pi^+_{\text{old}}(\cdot|\mathbf{x}),\mathbf{y}'\sim\pi^-_{\text{old}}(\cdot|\mathbf{x})}\ell(s(\mathbf{w};\mathbf{y}'\mathbf{x}) - s(\mathbf{w};\mathbf{y},\mathbf{x})). \tag{6.81}$$

[1] in the context of generative models, "predicts" is like "generates".

One difference from the objective of GRPO is that we use a single scoring function $s(\mathbf{w}; \mathbf{y}, \mathbf{x})$ for both positive outputs $\mathbf{y}$ and negative outputs $\mathbf{y}'$. The different scoring functions for positive and negative outputs in GRPO actually arise from the clipping operations. The clipping could cause the vanishing gradient, which may also slow down the learning process. To avoid these issues, we consider non-clipping scoring functions.

One advantage of designing the objective based on the principle of discriminative learning is the ability to leverage a wide range of advanced objectives to improve training. A key challenge in RL fine-tuning for reasoning models is the sparse rewards, which leads to imbalance in generated outputs. Specifically, for some questions where $p(\mathbf{x}) \ll 1$, the number of negative outputs can significantly exceed the number of positive ones. The objective function $\mathcal{L}_1$ is motivated by maximizing AUC for each question $\mathbf{x}$, i.e., $\mathbb{E}_{\mathbf{y}\sim\pi^+_{\text{old}},\mathbf{y}'\sim\pi^-_{\text{old}}}[\mathbb{I}(s(\mathbf{w};\mathbf{y},\mathbf{x}) > s(\mathbf{w};\mathbf{y}'\mathbf{x}))]$. However, when there is much more negative data than positive data, AUC is not a good measure. For example, let us consider a scenario that there are 1 positive $\mathbf{y}_+$ and 100 negatives $\{\mathbf{y}^1_-, \ldots, \mathbf{y}^{100}_-\}$. If the scores of these data are $s(\mathbf{y}^1_-, \mathbf{x}) = 0.9, s(\mathbf{y}_+, \mathbf{x}) = 0.5, s(\mathbf{y}^2_-, \mathbf{x}) = s(\mathbf{y}^3_-, \mathbf{x}) \ldots = s(\mathbf{y}^{100}_-, \mathbf{x}) = 0.001$, then the AUC score is $\frac{99}{100} = 0.99$. The AUC score is high but is not informative as the model still generates the negative data $\mathbf{y}^1_-$ more likely than the positive data $\mathbf{y}_+$.

To address this issue, we leverage the pAUC objective (6.28), leading to the following objective for minimization:

$$\mathcal{L}_2(\mathbf{w}) := \mathbb{E}_{\mathbf{x}}\mathbb{E}_{\mathbf{y}\sim\pi^+_{\text{old}}(\cdot|\mathbf{x})}\tau \log\left(\mathbb{E}_{\mathbf{y}'\sim\pi^-_{\text{old}}(\cdot|\mathbf{x})}\exp\left(\frac{\ell(s(\mathbf{w};\mathbf{y}',\mathbf{x}) - s(\mathbf{w};\mathbf{y},\mathbf{x}))}{\tau}\right)\right). \tag{6.82}$$

Lemma 2.4 indicates that $\mathcal{L}_2(\mathbf{w}) \geq \mathcal{L}_1(\mathbf{w})$ by Jensen's inequality for the concave function log. Hence, minimizing $\mathcal{L}_2(\mathbf{w})$ will automatically decreasing $\mathcal{L}_1(\mathbf{w})$. However, the reverse is not true. This also explains why minimizing $\mathcal{L}_2(\mathbf{w})$ could be more effective than maximizing $\mathcal{L}_1(\mathbf{w})$.

Scoring functions

Different scoring functions can be considered. Two examples are given below.

- The log-likelihood (log-L) scoring function is defined by

$$s(\mathbf{w}; \mathbf{y}, \mathbf{x}) = \frac{1}{|\mathbf{y}|}\sum_{t=1}^{|\mathbf{y}|}\log \pi_{\mathbf{w}}(y_t|\mathbf{x}, y_{<t}).$$

- The likelihood ratio (L-ratio) scoring function is computed by

$$s(\mathbf{w}; \mathbf{y}, \mathbf{x}) = \frac{1}{|\mathbf{y}|}\sum_{t=1}^{|\mathbf{y}|}\frac{\pi_{\mathbf{w}}(y_t|\mathbf{x}, y_{<t})}{\pi_{\text{old}}(y_t|\mathbf{x}, y_{<t})}.$$

Stabilize the training with Constrained Optimization

Training instability is a long-standing issue in RL. Instead of using the clipping operation of PPO, an effective approach is to use the idea of trust region constraint of TRPO, which restricts the updated model $\mathbf{w}$ in the trust region using the reverse KL:

$$\mathrm{KL}(\pi_{\text{old}}, \pi_{\mathbf{w}}) \leq \delta.$$

Putting It All Together

DisCO formulates policy learning as a discriminative constrained optimization problem that combines discriminative objectives with a trust-region constraint. Specifically, it solves one of the following two formulations:

$$\begin{aligned} &\min_{\mathbf{w}} \mathcal{L}_1(\mathbf{w}) \\ &\text{s.t.} \quad \mathrm{KL}(\pi_{\text{old}}, \pi_{\mathbf{w}}) \leq \delta, \end{aligned} \tag{6.83}$$

or alternatively,

$$\begin{aligned} &\min_{\mathbf{w}} \mathcal{L}_2(\mathbf{w}) \\ &\text{s.t.} \quad \mathrm{KL}(\pi_{\text{old}}, \pi_{\mathbf{w}}) \leq \delta. \end{aligned} \tag{6.84}$$

Optimization Algorithm

To tackle the constrained optimization, we can use the penalty method presented in next section, which converts the constrained problem into an unconstrained one with an appropriate penalty parameter β. For example, with a squared hinge penalty function, we solve

$$\min_{\mathbf{w}} \mathcal{L}(\mathbf{w}) + \beta[\mathrm{KL}(\pi_{\text{old}}, \pi_{\mathbf{w}}) - \delta]_+^2, \tag{6.85}$$

where $[\cdot]_+ = \max\{\cdot, 0\}$. We will show that under an appropriate assumption regarding the constraint function and β, solving the above squared-hinge penalized objective (6.85) can return a KKT solution of the original constrained problem (6.83).

We discuss the difference between using the squared-hinge penalty function and the regular KL divergence regularization $\beta\mathrm{KL}(\pi_{\text{old}}, \pi_\theta)$. The squared-hinge penalty function has a dynamic weighting impact for the gradient, $\nabla\beta[\mathrm{KL}(\pi_{\text{old}}, \pi_{\mathbf{w}}) - \delta]_+^2 = 2\beta[\mathrm{KL}(\pi_{\text{old}}, \pi_{\mathbf{w}}) - \delta]_+\nabla\mathrm{KL}(\pi_{\text{old}}, \pi_{\mathbf{w}})$, such that if the constraint is satisfied then the weight $2\beta[\mathrm{KL}(\pi_{\text{old}}, \pi_{\mathbf{w}}) - \delta]_+$ before the gradient of the regularization term $\mathrm{KL}(\pi_{\text{old}}, \pi_{\mathbf{w}})$ becomes zero. This means the KL divergence is only effective when the constraint is violated. In contrast, the regular KL divergence regularization $\beta\mathrm{KL}(\pi_{\text{old}}, \pi_{\mathbf{w}})$ always contributes a gradient $\beta\nabla\mathrm{KL}(\pi_{\text{old}}, \pi_{\mathbf{w}})$ no matter whether the constraint is satisfied or not, which could harm the learning.

The effectiveness of DisCO over GRPO and other methods has been demonstrated in (Li et al., 2025) for fine-tuning distilled Qwen and LLaMA models on a mathe-

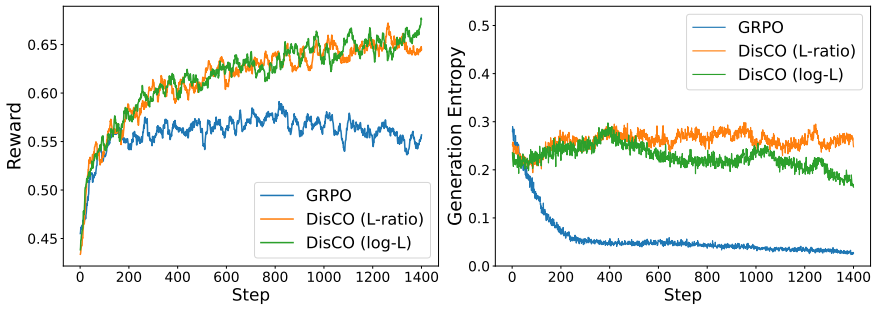


Fig. 6.29: Comparison of DisCO and GRPO for finetuning a 1.5B distilled Qwen model: left plots the training reward (averaged over generated outputs for questions used in each step) vs the number of training steps; right plots the generation entropy vs training steps. Each training step uses 128 questions sampled from the dataset, each associated with 8 generated responses to define the objective, and a mini-batch size of 32 is used for updates for a epoch. For more details, please refer to (Li et al., 2025).

matical reasoning data with approximately 40.3k unique problem-answer pairs. A comparison of the training dynamics for different methods is shown in Figure 6.29.

A PyTorch implementation of DisCO is included in the following Github repository:

https://github.com/Optimization-AI/DisCO.

6.7 Constrained Learning

Constrained learning is a machine learning framework in which the model is trained not only to minimize a specified risk but also to satisfy additional constraints. These constraints can encode domain knowledge, prior information, regularization terms, or other application-specific requirements. Unlike simple domain constraints $\mathbf{w} \in \mathcal{W}$, we consider complicated functional constraints in the form:

$$\begin{aligned} \min_{\mathbf{w}\in\mathbb{R}^d} \quad & F(\mathbf{w}) \\ s.t. \quad & g_i(\mathbf{w}) \leq 0,\ i = 1, \ldots, m. \end{aligned} \tag{6.86}$$

In many cases, $g_i(\mathbf{w})$ also depends on the data, making its evaluation and gradient computation expensive.

Traditional works for constrained optimization include three primary categories: (1) primal methods, e.g., cooperative subgradient methods and level-set methods; (2) primal-dual methods that reformulate constrained optimization problems as saddle point problems; (3) penalty-based approaches that incorporate constraints by adding a penalty term to the objective function. In this section, we demonstrate how FCCO enables penalty-based approaches to be both efficient and practically effective.

6.7.1 A General Penalty-based Approach via FCCO

To tackle the constraints, a penalty-based approach uses a penalty function $f(\cdot)$ to convert the constrained problem into an unconstrained one:

$$\min_{\mathbf{w}} F(\mathbf{w}) + \frac{\rho}{m} \sum_{i=1}^{m} f(g_i(\mathbf{w})), \tag{6.87}$$

where $\rho > 0$ is called the *penalty parameter*. Commonly used penalty functions include:

- Squared hinge penalty:
$$f(g) = \frac{1}{2}[g]_+^2,$$
- Hinge penalty:
$$f(g) = [g]_+,$$
- Smoothed hinge penalty:
$$f(g) = \begin{cases} g - \frac{\epsilon}{2} & \text{if } g \geq \epsilon, \\ \frac{g^2}{2\epsilon} & \text{if } 0 < g < \epsilon, \\ 0 & \text{otherwise,} \end{cases}$$
where $\epsilon \ll 1$ is a small constant.

Different penalty functions yield different convergence rates. However, they share a common property: when the constraints are satisfied at a point $\mathbf{w}$, no penalty is incurred; otherwise, the greater the violation, the larger the penalty.

We can see that the added second term in (6.87) is a form of FCCO. Hence, the algorithms developed in Chapter 5 can be applied to solving the resulting unconstrained problem. Nevertheless, we need to answer several important questions: (1) What is an appropriate value for ρ? (2) What convergence guarantees can be established for the original constrained problem?

Equivalent min-max formulation

By using the conjugate of f, the unconstrained problem is equivalent to:

$$\min_{\mathbf{w}} \max_{\mathbf{y} \in \mathrm{dom}^m(f^*)} F(\mathbf{w}) + \rho \frac{1}{m} \sum_{i=1}^{m} \left(y_i g_i(\mathbf{w}) - f^*(y_i)\right), \tag{6.88}$$

For the three penalty functions, we have

- Squared hinge penalty: $f^*(y) = \frac{1}{2}y^2$, $\mathrm{dom}(f^*) = \{y : y \geq 0\}$;
- Hinge penalty: $f^*(y) = \mathbb{I}_{0,\infty}[y \in \mathrm{dom}(f^*)]$, $\mathrm{dom}(f^*) = \{y : y \in [0, 1]\}$;

- Smoothed hinge penalty: $f^*(y) = \frac{\epsilon}{2}y^2$, $\text{dom}(f^*) = \{y : y \in [0, 1]\}$;

KKT solutions

Let us focus on non-convex optimization problems with a non-convex objective $F(\mathbf{w})$ and non-convex constraints $g_k(\mathbf{w}), \forall k$. For a non-convex optimization problem, finding a globally optimal solution is intractable. Instead, a Karush-Kuhn-Tucker (KKT) solution is of interest, which is an extension of a stationary solution of an unconstrained non-convex optimization problem.

Definition 6.1 (KKT solution) A solution $\mathbf{w}$ is a KKT solution to (6.86) if there exists $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_m)^\top \in \mathbb{R}^m_+$ such that (i) $0 \in \partial F(\mathbf{w}) + \sum_{k=1}^m \lambda_k \partial g(\mathbf{w})$, (ii) $g_k(\mathbf{w}) \leq 0, \forall k$ and (iii) $\lambda_k g_k(\mathbf{w}) = 0, \forall k$.

For non-asymptotic analysis, we consider finding an ϵ-KKT solution as defined below.

Definition 6.2 A solution $\mathbf{w}$ is an ϵ-KKT solution to (6.86) if there exists $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_m)^\top \in \mathbb{R}^m_+$ such that (i): $\text{dist}(0, \partial F(\mathbf{w}) + \sum_{k=1}^m \lambda_k \partial g_k(\mathbf{w})) \leq \epsilon$, (ii): $[g_k(\mathbf{w})]_+ \leq \epsilon, \forall k$, and (iii): $|\lambda_k g_k(\mathbf{w})| \leq \epsilon, \forall k$.

If the objective and the constraint functions are non-smooth, finding an ϵ-KKT solution is not tractable, even the constraint functions are absent. For example, if $F(x) = |x|$ finding ϵ-stationary solution is infeasible unless we find the optimal solution $x = 0$. To address this challenge, we consider finding a nearly ϵ-KKT solution defined below.

Definition 6.3 A solution $\mathbf{w}$ is a nearly ϵ-KKT solution to (6.86) if there exist $\bar{\mathbf{w}}$ and $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_m)^\top \in \mathbb{R}^m_+$ such that (i): $\|\mathbf{w} - \bar{\mathbf{w}}\|_2 \leq O(\epsilon)$, $\text{dist}(0, \partial F(\bar{\mathbf{w}}) + \sum_{k=1}^m \lambda_k \partial g_k(\bar{\mathbf{w}})) \leq \epsilon$, (ii): $[g_k(\bar{\mathbf{w}})]_+ \leq \epsilon, \forall k$, and (iii): $|\lambda_k g_k(\bar{\mathbf{w}})| \leq \epsilon, \forall k$.

Theory

Solving the unconstrained problem (6.87) can yield a (nearly) stationary solution. But is this solution close to satisfying the KKT conditions of the original constrained problem? We answer this question for the three penalty functions below. Let $\boldsymbol{g}(\mathbf{w}) = (g_1(\mathbf{w}), \ldots, g_m(\mathbf{w}))^\top \in \mathbb{R}^m$ denote the vector of constraint functions, and let $\nabla \boldsymbol{g}(\mathbf{w}) \in \mathbb{R}^{m \times d}$ denote its Jacobian matrix.

Squared Hinge Penalty

Let us assume F and g_k are differentiable. We make the following assumption regarding the regularity of the constraint functions.

Assumption 6.1. *There exists a constant $\delta > 0$ such that $\sigma_{\min}(\nabla \boldsymbol{g}(\mathbf{w})) \geq \delta$ for any $\mathbf{w}$ satisfying $\max_{k=1,\ldots,K} g_k(\mathbf{w}) > 0$, where $\sigma_{\min}(\cdot)$ denotes the minimum singular value of a matrix.*

This assumption implies that when any constraint is violated, its gradient direction can be used to effectively reduce the constraint value. To illustrate this, consider a single constraint defined by a L_g-smooth function $g(\cdot)$. Suppose $\mathbf{w}$ is a point where the constraint is violated, i.e., $g(\mathbf{w}) > 0$. Taking a gradient descent step $\mathbf{w}' = \mathbf{w} - \eta \nabla g(\mathbf{w})$ yields:

$$\begin{aligned} g(\mathbf{w}') &\leq g(\mathbf{w}) + \nabla g(\mathbf{w})^\top (\mathbf{w}' - \mathbf{w}) + \frac{L_g}{2} \|\mathbf{w}' - \mathbf{w}\|_2^2 \\ &= g(\mathbf{w}) - \left(\eta - \frac{L_g \eta^2}{2}\right) \|\nabla g(\mathbf{w})\|_2^2. \end{aligned}$$

If Assumption 6.1 holds, then $\|\nabla g(\mathbf{w})\|_2 \geq \delta$, which implies:

$$g(\mathbf{w}') \leq g(\mathbf{w}) - \left(\eta - \frac{L_g \eta^2}{2}\right) \delta^2,$$

ensuring a sufficient decrease in the constraint function value.

In addition, we need to assume the objective function is Lipschitz continuous.

Assumption 6.2. *There exists a constant $C > 0$ such that $\|\nabla F(\mathbf{w})\|_2 \leq C, \forall \mathbf{w}$.*

Under these assumptions, we establish the following theorem.

Theorem 6.1 *Suppose Assumption 6.1 and 6.2 hold. Let $\mathbf{w}$ be an ϵ-stationary solution to the unconstrained penalized problem (6.87) with a squared hinge penalty such that*

$$\mathbb{E}\left[\left\|\nabla F(\mathbf{w}) + \frac{\rho}{m} \nabla \boldsymbol{g}(\mathbf{w})^\top [\boldsymbol{g}(\mathbf{w})]_+\right\|_2^2\right] \leq \epsilon^2. \tag{6.89}$$

If $\rho \geq \max(\frac{2m(C^2+1)}{\epsilon \delta^2}, \frac{m\sqrt{2(C^2+1)}}{\epsilon \delta})$, then $\mathbf{w}$ is also an ϵ-KKT solution to the original problem (6.86).

Proof. Let $\lambda_k = \frac{\rho}{m}[g_k(\mathbf{w})]_+, \forall k$. If $\max_k g_k(\mathbf{w}) \leq 0$, then $\lambda_k = 0$. As a result, $\mathbf{w}$ is an ϵ-KKT solution to the original problem.

Below, let us focus on the case $\max_k g_k(\mathbf{w}) > 0$, i.e., there exists one constraint that is violated at $\mathbf{w}$. Then, under Assumption 6.1, we have

$$
\begin{aligned}
\|[\boldsymbol{g}(\mathbf{w})]_+\|_2^2 \leq & \frac{1}{\delta^2}\|\nabla \boldsymbol{g}(\mathbf{w})^\top [\boldsymbol{g}(\mathbf{w})]_+\|_2^2 \\
= & \frac{m^2}{\rho^2\delta^2}\left\|\nabla F(\mathbf{w}) + \frac{\rho}{m}\nabla \boldsymbol{g}(\mathbf{w})^\top [\boldsymbol{g}(\mathbf{w})]_+ - \nabla F(\mathbf{w})\right\|_2^2 \\
\leq & \frac{2m^2}{\rho^2\delta^2}\left[\|\nabla F(\mathbf{w})\|_2^2 + \left\|\nabla F(\mathbf{w}) + \frac{\rho}{m}\nabla \boldsymbol{g}(\mathbf{w})[\boldsymbol{g}(\mathbf{w})]_+\right\|_2^2\right] \\
\leq & \frac{2m^2}{\rho^2\delta^2}\left[C^2 + \epsilon^2\right] \leq \epsilon^2,
\end{aligned}
\tag{6.90}
$$

where the last inequality follows from $\rho \geq \frac{m\sqrt{2(C^2+\epsilon^2)}}{\delta\epsilon}$. Hence $[g_k(\mathbf{w})]_+ \leq \epsilon, \forall k$.

Then, let us bound $|\lambda_k g_k(\mathbf{w})|$. If $g_k(\mathbf{w}) < 0$, then $\lambda_k = 0$, we have $|\lambda_k g_k(\mathbf{w})| = 0$. If $g_k(\mathbf{w}) \geq 0$, then

$$
\begin{aligned}
\mathbb{E}|\lambda_k g_k(\mathbf{w})]| = \mathbb{E}|\frac{\rho}{m} g_k(\mathbf{w}) g_k(\mathbf{w})]| \leq & \frac{\rho}{m}\mathbb{E}\|[\boldsymbol{g}(\mathbf{w})]_+\|_2^2 \\
\leq & \frac{\rho}{m}\cdot\frac{2m^2}{\rho^2\delta^2}\left[\|\nabla F(\mathbf{w})\|_2^2 + \left\|\nabla F(\mathbf{w}) + \frac{\rho}{m}\nabla \boldsymbol{g}(\mathbf{w})[\boldsymbol{g}(\mathbf{w})]_+\right\|_2^2\right] \\
\leq & \frac{2m}{\rho\delta^2}\left[C^2 + \epsilon^2\right] \leq \epsilon
\end{aligned}
\tag{6.91}
$$

where the last inequality uses $\rho \geq \frac{2m(C^2+\epsilon^2)}{\epsilon\,\delta^2}$. □

Hinge Penalty

Since the hinge function is non-smooth, let us consider non-smooth F and g_k. We make the following assumption regarding the regularity of the constraint functions.

Assumption 6.3. *There exists a constant $\delta > 0$ such that*

$$
dist\left(0, \frac{1}{m}\sum_{k=1}^{m}\partial[g_k(\mathbf{w})]_+\right) \geq \frac{\delta}{m}, \forall \mathbf{w} \in \mathcal{V} \tag{6.92}
$$

where $\mathcal{V} = \{\mathbf{w} : \max_k g_k(\mathbf{w}) > 0\}$ and $\partial[g_k(\mathbf{w})]$ denotes the subgradient in terms of $\mathbf{w}$.

The above assumption is implied by Assumption 6.1 when g is differentiable and hence is weaker. To see this, we have

$$
\text{dist}\left(0, \frac{1}{m}\sum_{k=1}^{m}\nabla[g_k(\mathbf{w})]_+\right) = \left\|\frac{1}{m}\sum_{k=1}^{m}\nabla[g_k(\mathbf{w})]_+\right\|_2 = \|\nabla g(\mathbf{w})^\top \mathbf{a}\|_2 \geq \delta\|\mathbf{a}\|_2 \geq \frac{\delta}{m},
$$

where $\mathbf{a} = \frac{1}{m}(\xi_1, \ldots, \xi_m)$, and $\xi_k \in ([g_k(\mathbf{w})]_+)' \in [0, 1]$.

Theorem 6.2 *Suppose Assumption 6.3 and Assumption 6.2 hold. Let* $\mathbf{w}$ *be a nearly ϵ-stationary solution to the unconstrained penalized problem (6.87) with a hinge*

penalty such that there exists $\bar{\mathbf{w}}$ *satisfying* $\|\mathbf{w} - \bar{\mathbf{w}}\|_2 \leq O(\epsilon)$*, and*

$$dist\left(0, \partial F(\bar{\mathbf{w}}) + \frac{\rho}{m}\sum_{k=1}^{m} \partial [g_k(\bar{\mathbf{w}})]_+\right) \leq \epsilon.$$

If $\rho > \frac{m(C+1)}{\delta}$*, then* $\mathbf{w}$ *is a nearly* ϵ*-KKT solution to the original problem (6.86).*

Proof. By the definition of $\mathbf{w}$, there exists $\bar{\mathbf{w}}$ such that $\|\mathbf{w} - \bar{\mathbf{w}}\|_2 \leq O(\epsilon)$, and

$$\text{dist}\left(0, \partial F(\bar{\mathbf{w}}) + \frac{\rho}{m}\sum_{k=1}^{m} \partial [g_k(\bar{\mathbf{w}})]_+\right) \leq \epsilon.$$

Since $\partial [g_k(\bar{\mathbf{w}})]_+ = \xi_k \partial g_k(\bar{\mathbf{w}})$, where

$$\xi_k = \begin{cases} 1 & \text{if } g_k(\bar{\mathbf{w}}) > 0, \\ [0,1] & \text{if } g_k(\bar{\mathbf{w}}) = 0, \\ 0 & \text{if } g_k(\bar{\mathbf{w}}) < 0, \end{cases} \quad \in [g_k(\bar{\mathbf{w}})]'_+,$$

there exists $\lambda_k \in \frac{\rho \xi_k}{m} \geq 0, \forall k$ such that

$$\text{dist}\left(0, \partial F(\bar{\mathbf{w}}) + \sum_{k=1}^{m} \lambda_k \partial g_k(\bar{\mathbf{w}})\right) \leq \epsilon.$$

Thus, we prove condition (i) in Definition 6.3. Next, let us prove condition (ii). We argue that $\max_k g_k(\bar{\mathbf{w}}) \leq 0$. Suppose this does not hold, i.e., $\max_k g_k(\bar{\mathbf{w}}) > 0$, we will derive a contradiction. Since $\exists \mathbf{v} \in \partial F(\bar{\mathbf{w}})$ we have

$$\begin{aligned} \epsilon &\geq \text{dist}\left(0, \mathbf{v} + \frac{\rho}{m}\sum_{k=1}^{m} \partial [g_k(\bar{\mathbf{w}})]_+\right) \\ &\geq \text{dist}\left(0, \frac{\rho}{m}\sum_{k=1}^{m} \partial [g_k(\bar{\mathbf{w}})]_+\right) - \|\mathbf{v}\|_2 \geq \frac{\rho\delta}{m} - C, \end{aligned}$$

which is a contradiction to the assumption that $\rho > \frac{m(\epsilon+C)}{\delta}$. Thus, $\max_k g_k(\bar{\mathbf{x}}) \leq 0$. This proves condition (ii). The last condition (iii) holds because: $\lambda_k = \frac{\rho \xi_k}{m}$, which is zero if $g_k(\bar{\mathbf{w}}) < 0$. Hence, $\lambda_k g_k(\bar{\mathbf{w}}) = 0$. □

Smoothed Hinge Penalty

We make the following assumption regarding the regularity of the constraint functions.

Assumption 6.4. *There exists a constant* $\delta > 0$ *such that*

$$dist\left(0, \partial g(\mathbf{w})^\top \mathbf{v}\right) \geq \delta \|\mathbf{v}\|_2, \forall \mathbf{w} \in \mathcal{V}, \forall \mathbf{v} \in \mathbb{R}^m \tag{6.93}$$

where $\mathcal{V} = \{\mathbf{w} : \max_k g_k(\mathbf{w}) > 0\}$.

Theorem 6.3 *Suppose Assumption 6.1 and Assumption 6.2 hold. Let* $\mathbf{w}$ *be a nearly* ϵ*-stationary solution to the unconstrained penalized problem (6.87) with a smoothned hinge penalty such that there exists* $\bar{\mathbf{w}}$ *satisfying* $\|\mathbf{w} - \bar{\mathbf{w}}\|_2 \leq O(\epsilon)$*, and*

$$dist\left(0, \partial F(\bar{\mathbf{w}}) + \frac{\rho}{m}\sum_{k=1}^{m} \partial f(g_k(\bar{\mathbf{w}}))\right) \leq \epsilon.$$

If $\rho > \frac{m(C+1)}{\delta}$*, then there exists* $\boldsymbol{\lambda} \in \mathbb{R}_+^m$ *it holds (i)* $\|\mathbf{w}-\bar{\mathbf{w}}\| \leq O(\epsilon)$*,* $dist(0, \partial F(\bar{\mathbf{w}}) + \sum_{k=1}^m \lambda_k \partial g_k(\bar{\mathbf{w}})) \leq \epsilon$*, (ii)* $[g_k(\bar{\mathbf{w}})]_+ \leq \epsilon, \forall k$*, and (iii)* $\lambda_k [g_k(\bar{\mathbf{w}})]_+ \leq \rho\epsilon/m, \forall k$*.*

Proof. By the definition of $f(\cdot)$, we have

$$\nabla f(\cdot) = \frac{1}{\epsilon}\min\{[\cdot]_+, \epsilon\}.$$

According to the definition of $\mathbf{w}$, there exists $\bar{\mathbf{w}}$ such that $\|\mathbf{w} - \bar{\mathbf{w}}\|_2 \leq O(\epsilon)$ and

$$\text{dist}\left(0, \partial F(\bar{\mathbf{w}}) + \frac{\rho}{m}\sum_{k=1}^{m} \nabla f[g_k(\bar{\mathbf{w}})]\partial g_k(\bar{\mathbf{w}})\right) \leq \epsilon.$$

Let $\lambda_k = \frac{\rho}{m}\nabla f(g_i(\bar{\mathbf{w}})) = \frac{\rho}{\epsilon m}\min\{[g_k(\bar{\mathbf{w}})]_+, \epsilon\}$. Then,

$$\text{dist}\left(0, \partial F(\bar{\mathbf{w}}) + \sum_{k=1}^{m} \lambda_k \partial g_k(\bar{\mathbf{w}})\right) \leq \epsilon.$$

Suppose $\max_{i=1,\ldots,m} g_i(\bar{\mathbf{w}}) > \epsilon$. Then there exists k' such that $[g_k(\bar{\mathbf{w}})]_+ > \epsilon$. Hence

$$\lambda_{k'} = \frac{\rho}{\epsilon m}\min\{[g_{k'}(\bar{\mathbf{w}})]_+, \epsilon\} = \frac{\rho}{\epsilon m}\epsilon = \frac{\rho}{m}.$$

Hence $\|\boldsymbol{\lambda}\|_2 \geq \frac{\rho}{m}$. As a result, there exists $\mathbf{v} \in \partial F(\bar{\mathbf{w}})$ such that

$$\begin{aligned}\epsilon &\geq \text{dist}\left(0, \mathbf{v} + \sum_{k=1}^{m} \lambda_k \partial g_k(\bar{\mathbf{w}})\right) \\ &\geq \text{dist}\left(0, \sum_{k=1}^{m} \lambda_k \partial g_k(\bar{\mathbf{w}})\right) - \|\mathbf{v}\|_2 \geq \frac{\rho\delta}{m} - C, \end{aligned} \tag{6.94}$$

which contradicts with $\rho > \frac{m(C+\epsilon)}{\delta}$. Therefore, we must have

$$\max_{k=1,\ldots,m} g_k(\bar{\mathbf{w}}) \leq \epsilon. \tag{6.95}$$

Finally, let us prove $|\lambda_k g_k(\bar{\mathbf{w}})| \leq O(\epsilon)$. If $g_k(\bar{\mathbf{w}}) < 0$, we have $\lambda_k = 0$, then it holds trivially. If $0 \leq g_k(\bar{\mathbf{w}}) \leq \epsilon$, we have

Algorithm	Penalty	F	g_i	Complexity	Loop
SOX	sqH/smH	SM	SM	$O(\epsilon^{-7})$	Single
MSVR	sqH/smH	MSS	MSS	$O(\epsilon^{-5})$	Single
SONX	H	WC	WC	$O(\epsilon^{-6})$	Single
SONEX	H	SM	SM	$O(\epsilon^{-5})$	Single
ALEXR-DL	smH	WC	WC	$O(\epsilon^{-5})$	Double

Table 6.1: Summary of different algorithms for penalty-based constrained optimization. 'WC" means weakly convex, "SM" means smooth, MSS mean "mean squared smoothness, 'H' denotes the hinge penalty, 'smH' denotes the smoothed hinge penalty and 'sqH' denotes the squared hinge penalty.

$$|\lambda_k g_k(\bar{\mathbf{w}})| \leq \frac{\rho}{m}[g_k(\bar{\mathbf{w}})]_+ \leq \frac{\rho\epsilon}{m}. \tag{6.96}$$

□

Critical: One important difference among the three penalty functions lies in the required order of the penalty parameter ρ. For the squared hinge penalty, it is necessary to set $\rho = O(1/\epsilon)$, whereas for the hinge and smoothed hinge penalties, it suffices to take $\rho = O(1)$. This lead to different complexities of algorithms based on these penalty functions.

Optimization Algorithms

The SOX algorithm and the MSVR algorithm can be used to optimize the squared hinge penalty function and smoothed hinge penalty function with smooth objective function and constraints. SONX and SONEX can be used to optimize the hinge penalty based objective, where the latter is equivalent to a variant for optimizing the smoothed hinge penalty using the MSVR estimator for the inner functions and the MA gradient estimator. ALEXR-DL (the double-loop ALEXR, see Section 5.4.5) can be used to optimize the problem with a weakly convex objective and weakly convex constraint functions. The computational complexities of these algorithms for obtaining a (nearly) ϵ-KKT solution are summarized in Table 6.1. The complexity results for SONX and SONEX follow directly from their original theorems. The complexities of SOX and MSVR are obtained by substituting $L_F = O(\rho)$, $L_1 = O(\rho)$, $G_1 = O(\rho)$, and $\rho = O(1/\epsilon)$ into Theorem 5.1 and Theorem 5.2, respectively. The complexity of ALEXR-DL follows the argument in Section 5.4.5.

Finally, we note that the value of the parameter δ in Assumptions 6.1, 6.3, and 6.4 has a significant impact on the complexity. In particular, smaller values of δ lead to higher complexities.

6.7.2 Continual Learning with Zero-forgetting Constraints

Continual learning usually refers to learning a sequence of tasks one by one and accumulating knowledge like human instead of substituting knowledge. The core issue in continual learning is known as catastrophic forgetting, i.e., the learning of the later tasks may significantly degrade the performance of the model for the earlier tasks. Different approaches have been investigated to mitigate catastrophic forgetting, including regularization based approaches, memory based approaches, network expansion based approaches, and constrained optimization based approaches.

Regularization based approaches

These methods aim to preserve previously learned knowledge by penalizing changes to important model parameters. These approaches usually solve the following objective:

$$\min_{\mathbf{w}} \mathcal{L}_{\text{new}}(\mathbf{w}, \mathcal{S}_{\text{new}}) + \lambda R(\mathbf{w}, \mathbf{w}_{\text{old}}), \tag{6.97}$$

where $\mathcal{L}_{\text{new}}$ denotes the loss on the new task with a data set $\mathcal{S}_{\text{new}}$, and $R(\mathbf{w}, \mathbf{w}_{\text{old}})$ is the regularization of the new model with respect to the old model. It could regularize directly in the weight parameters or regularize through functions of the weight parameters (e.g., intermediate layers of the neural networks)

Memory based approaches

These techniques store a subset of past data or representations and replay them during training on new tasks. This allows the model to rehearse old knowledge, effectively mimicking how humans review what they've previously learned. Strategies include storing raw data, or using generative models to simulate past experiences. These replay data will be used in training as simple as a regularization approach:

$$\min_{\mathbf{w}} \mathcal{L}_{\text{new}}(\mathbf{w}, \mathcal{S}_{\text{new}}) + \lambda \mathcal{L}_{\text{old}}(\mathbf{w}, \mathcal{S}_{\text{old}}) \tag{6.98}$$

where $\mathcal{L}_{\text{old}}(\mathbf{w}, \mathcal{S}_{\text{old}})$ denotes the loss of the model old tasks using their data $\mathcal{S}_{\text{old}}$.

Network Expansion based approaches

Network expansion based methods address forgetting by dynamically growing the model's architecture as new tasks are introduced. This can involve adding new neurons, layers, or modules for each task while keeping older components fixed or partially shared. By allocating new capacity, the model can learn new tasks without overwriting old knowledge.

A Constrained Optimization Approach

A key limitation of the replay and regularization approach in (6.98) is that it does not necessarily preserve the model's performance on all previous tasks, even with a large regularization weight. Moreover, overly large weights can suppress learning on the new task. This arises because not all prior tasks are equally challenging—some may be inherently easier than others.

A straightforward remedy is to formulate a constrained optimization problem:

$$\begin{aligned} \min_{\mathbf{w}} \quad & \mathcal{L}_{\text{new}}(\mathbf{w}, \mathcal{S}_{\text{new}}) \\ \text{s.t.} \quad & \mathcal{L}_k(\mathbf{w}, \mathcal{S}_k) - \mathcal{L}_k(\mathbf{w}_{\text{old}}, \mathcal{S}_k) \leq 0, \quad \forall k = 1, \ldots, m, \end{aligned} \tag{6.99}$$

where $\mathcal{S}_k$ denotes the dataset for the k-th previous task and $\mathcal{L}_k$ is its corresponding loss function. These constraints ensure that the new model does not degrade performance on any individual old task as measured on replayed data, which are referred to as the zero-forgetting constraints.

Although this constrained optimization problem was traditionally considered difficult due to the number of constraints and data dependencies, the algorithms introduced in the previous subsection make it tractable. Notably, this constrained formulation serves as a unifying framework that connects all three major approaches: regularization-based, expansion-based, and memory-based continual learning.

With a penalty function f (e.g., smoothed hinge penalty), we solve the following problem:

$$\min_{\mathbf{w}} \quad \mathcal{L}_{\text{new}}(\mathbf{w}, \mathcal{S}_{\text{new}}) + \frac{\rho}{m} \sum_{k=1}^{m} f(\mathcal{L}_k(\mathbf{w}, \mathcal{S}_k) - \mathcal{L}_k(\mathbf{w}_{\text{old}}, \mathcal{S}_k)).$$

Then the algorithms can be easily applied to solving this problem.

Connection with the Three Categories of Approaches

First, the above constrained optimization method falls under memory based approaches, as it requires access to data $\mathcal{S}_k$ from each previous task to define the zero-forgetting constraints.

Second, the penalty term introduces a regularization perspective, establishing a connection with regularization based approaches. However, it differs from standard regularization as in (6.98). The penalty function adaptively weights the gradients of each prior task. For example, consider the hinge penalty. The gradient of the penalty term is given by

$$\frac{\rho}{m} \sum_{k=1}^{m} \xi_k \nabla \mathcal{L}_k(\mathbf{w}; \mathcal{S}_k), \tag{6.100}$$

where $\xi_k = 1$ if $\mathcal{L}_k(\mathbf{w}; \mathcal{S}_k) - \mathcal{L}_k(\mathbf{w}_{\text{old}}; \mathcal{S}_k) > 0$; otherwise, $\xi_k = 0$. Using the FCCO technique, an estimator u_k is used to track the quantity $\mathcal{L}_k(\mathbf{w}; \mathcal{S}_k) -$

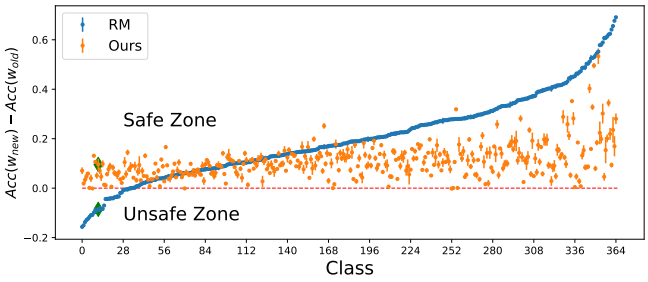


Fig. 6.30: Performance comparison with the standard regularization method (RM). The new task is to improve the performance on classifying the class *Dresssing Room* on Places365 Dataset, and other 354 classes serve as previous tasks each with 2k samples. Red line denotes the old model's performance, green diamonds denote the performance on the target class. The RM baseline shown is for the regularization parameter $\lambda = 10000$. For more details, please refer to (Li et al., 2024)

$\mathcal{L}_k(\mathbf{w}_{\text{old}}; \mathcal{S}_k)$, based on which ξ_k is computed. Consequently, the algorithm assigns adaptive weights to the gradients of prior tasks: if task k shows no performance degradation (i.e., $u_k \leq 0$), the corresponding gradient receives zero weight. This effect makes the constrained optimization approach more attractive than the regularization approach for enforcing the constraints; see Figure 6.30.

Third, although the connection to network expansion based approaches is less direct, it is suggested by the convergence analysis of the constrained optimization algorithms. Specifically, the regularity assumptions in Assumptions 6.1 and 6.3 provide insight into the benefits of network expansion. Expanding the network from the old model $\mathbf{w}_{\text{old}}$ can make it easier to find a new model that maintains or improves performance on previous tasks, effectively increasing the regularity constant δ. This, in turn, allows for a smaller penalty parameter ρ and potentially accelerates convergence—an effect formalized in what follows.

Without causing confusion, we denote by $\mathbf{w}$ the parameter of the old neural network, which consists of two components $\mathbf{w}_0$ and W such that the output $h(\mathbf{w}, \mathbf{x}) \in \mathbb{R}^{d_2}$ can be represented as $h(\mathbf{w}, \mathbf{x}) = W \cdot h_0(\mathbf{w}_0, \mathbf{x})$, where $h_0(\mathbf{w}_0, \cdot) \in \mathbb{R}^{d_1}$ is a backbone network and $W \in \mathbb{R}^{d_2 \times d_1}$ is the head. Given the old model $\mathbf{w} = (\mathbf{w}_0, W)$, we expand the network by allowing task-dependent heads, which is to let each task k have its own head $W_k = W + U_k$ where $U_k \in \mathbb{R}^{d_2 \times r}$. The output of this expanded network for task k is $h(\hat{\mathbf{w}}; \mathbf{x}) = (W + U_k) \cdot h_0(\mathbf{w}_0, \mathbf{x})$, where $\hat{\mathbf{w}} = (\mathbf{w}_0, W, U_1, \ldots, U_m)$. For simplicity, let us assume each task has only one example $\mathcal{S}_k = \{\mathbf{x}_k\}$ and let $\mathcal{L}_k(\mathbf{w}; \mathbf{S}_k) = \ell(h(\mathbf{w}, \mathbf{x}))$. Without the expansion, the Jacobian of the constraint functions at $\mathbf{w}$ is $\nabla \boldsymbol{g}(\mathbf{w}) = [\nabla h(\mathbf{w}, \mathbf{x}_1), \cdots, \nabla h(\mathbf{w}, \mathbf{x}_m)]A$, where $A \in \mathbb{R}^{m \times m}$ a diagonal matrix with $A_{kk} = \ell'(h(\mathbf{w}; \mathbf{x}_k))$. With the expansion, the Jacobian of the constraint functions at $\hat{\mathbf{w}}$ is $\nabla \hat{\boldsymbol{g}}(\mathbf{w}) = [\nabla h(\hat{\mathbf{w}}, \mathbf{x}_1), \cdots, \nabla h(\hat{\mathbf{w}}, \mathbf{x}_m)]A'$, where $A' \in \mathbb{R}^{m \times m}$ a diagonal matrix with $A'_{kk} = \ell'(h(\hat{\mathbf{w}}; \mathbf{x}_k))$. If we initialize $U_1 = U_2 \ldots = U_m = 0$, then $A = A'$. Next, we quantify the increase of the minimum singular value of

the matrix $\nabla\hat{\mathbf{h}}(\hat{\mathbf{w}}) = [\nabla h(\hat{\mathbf{w}}, \mathbf{x}_1), \cdots, \nabla h(\hat{\mathbf{w}}, \mathbf{x}_m)]$ compared with that of $\nabla\mathbf{h}(\mathbf{w}) = [\nabla h(\mathbf{w}, \mathbf{x}_1), \cdots, \nabla h(\mathbf{w}, \mathbf{x}_m)]$.

Lemma 6.4 *Suppose $U_k = \mathbf{0}$ for all k. We have*

$$\lambda_{\min}\left(\nabla\hat{\boldsymbol{h}}(\hat{\mathbf{w}})^\top \nabla\hat{\boldsymbol{h}}(\hat{\mathbf{w}})\right) \geq \lambda_{\min}\left(\nabla\boldsymbol{h}(\mathbf{w})^\top \nabla\boldsymbol{h}(\mathbf{w})\right) + \min_k \|\nabla_W h_k(\mathbf{w})\|_2^2,$$

where $\lambda_{\min}(\cdot)$ denotes the minimum eigen-value of a matrix and $h_k(\mathbf{w}) = h(\mathbf{w}; \mathbf{x}_k)$.

Why it matters

This lemma indicates that expanding the network can increase the minimum singular value of the Jacobian matrix of the constraint functions, which in turn leads to a lower complexity in finding a KKT solution, i.e., making the constraints easier to satisfy.

Proof. Let $\hat{h}_k(\hat{\mathbf{w}}) = h(\hat{\mathbf{w}}; \mathbf{x}_k)$. We consider $\mathbf{w}, W, U$ as flattend vectors. Recall that $\mathbf{w}$ has two component $\mathbf{w}_0$ and W. The gradient of $h_k(\mathbf{w})$ with respect to W and $\mathbf{w}_0$ are denoted by $\nabla_W h_k(\mathbf{w})$ and $\nabla_{\mathbf{w}_0} h_k(\mathbf{w})$, respectively. Hence,

$$\nabla h_k(\mathbf{w})^\top = \left(\nabla_{\mathbf{w}_0} h_k(\mathbf{w})^\top, \nabla_W h_k(\mathbf{w})^\top\right)$$

for $k = 1, \ldots, m$. Similarly, after adding the task-dependent heads, $\hat{\mathbf{w}}$ has three component $\mathbf{w}_0$, W, $\mathbf{U} = (U_1, \ldots, U_m)$. The gradients $\nabla_{\mathbf{w}_0}\hat{h}_k(\hat{\mathbf{w}})$, $\nabla_W \hat{h}_k(\hat{\mathbf{w}})$ $\nabla_{\mathbf{U}}\hat{h}_k(\hat{\mathbf{w}})$ are defined correspondingly, and

$$\nabla\hat{h}_k(\hat{\mathbf{w}})^\top = \left(\nabla_{\mathbf{w}_0}\hat{h}_k(\hat{\mathbf{w}})^\top, \nabla_W \hat{h}_k(\hat{\mathbf{w}})^\top, \nabla_{\mathbf{U}}\hat{h}_k(\hat{\mathbf{w}})^\top\right).$$

Recall that

$$\hat{h}_k(\hat{\mathbf{w}}) = h_k((\mathbf{w}_0, W + U_k)) \text{ for } k = 1, \ldots, m.$$

Therefore,

$$\nabla_{\mathbf{w}_0}\hat{h}_k(\hat{\mathbf{w}}) = \nabla_{\mathbf{w}_0} h_k((\mathbf{w}_0, W + U_k)),$$
$$\nabla_W \hat{h}_k(\hat{\mathbf{w}}) = \nabla_W h_k((\mathbf{w}_0, W + U_k)),$$

and

$$\nabla_{\mathbf{U}}\hat{h}_k(\hat{\mathbf{w}})^\top = \Big(\mathbf{0}, \ldots, \mathbf{0}, \underbrace{\nabla_W h_k((\mathbf{w}_0, \mathbf{W} + U_k))^\top}_{\text{The } k\text{th block}}, \mathbf{0}, \ldots, \mathbf{0}\Big),$$

where the sparsity pattern of $\nabla_{\mathbf{U}}\hat{h}_k(\hat{\mathbf{w}})$ is because $\hat{h}_k$ does not depend on $U_j, j \neq k$.

Since $U_k = \mathbf{0}$ for all k. It holds that $h_k(\mathbf{w}) = \hat{h}_k(\hat{\mathbf{w}})$ and

$$\nabla h_k(\mathbf{w})^\top = \left(\nabla_{\mathbf{w}_0} h_k(\mathbf{w})^\top, \nabla_W h_k(\mathbf{w})^\top\right) = \left(\nabla_{\mathbf{w}_0}\hat{h}_k(\hat{\mathbf{w}})^\top, \nabla_W \hat{h}_k(\hat{\mathbf{w}})^\top\right).$$

Consider any $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_m) \in \mathbb{R}^m$. We have

$$
\begin{aligned}
&\lambda_{\min}\left(\left[\nabla \hat{h}_1(\hat{\mathbf{w}}), \ldots, \nabla \hat{h}_m(\hat{\mathbf{w}})\right]^\top \left[\nabla \hat{h}_1(\hat{\mathbf{w}}), \ldots, \nabla \hat{h}_m(\hat{\mathbf{w}})\right]\right) \\
=&\min_{\boldsymbol{\alpha}, \text{s.t.} \|\boldsymbol{\alpha}\|=1} \left\| \sum_{k=1}^m \alpha_k \nabla \hat{h}_k(\hat{\mathbf{w}}) \right\|_2^2 \\
=&\min_{\boldsymbol{\alpha}, \text{s.t.} \|\boldsymbol{\alpha}\|=1} \left(\left\| \sum_{k=1}^m \alpha_k \nabla_{\mathbf{w}_0} \hat{h}_k(\hat{\mathbf{w}}) \right\|_2^2 + \left\| \sum_{k=1}^m \alpha_k \nabla_W \hat{h}_k(\hat{\mathbf{w}}) \right\|_2^2 + \left\| \sum_{k=1}^m \alpha_k \nabla_{\mathbf{U}} \hat{h}_k(\hat{\mathbf{w}}) \right\|_2^2 \right) \\
=&\min_{\boldsymbol{\alpha}, \text{s.t.} \|\boldsymbol{\alpha}\|=1} \left(\left\| \sum_{k=1}^m \alpha_k \nabla h_k(\mathbf{w}) \right\|_2^2 + \sum_{k=1}^m \alpha_k^2 \left\| \nabla_W h_k(\mathbf{w}) \right\|_2^2 \right) \\
\geq&\lambda_{\min}\left(\left[\nabla h_1(\mathbf{w}), \ldots, \nabla h_m(\mathbf{w})\right]^\top \left[\nabla h_1(\mathbf{w}), \ldots, \nabla h_m(\mathbf{w})\right] \right) \\
&+ \min_k \left\| \nabla_W h_k(\mathbf{w}) \right\|_2^2,
\end{aligned}
$$

where the first two equalities are by definitions and the third equality is because $U_k = 0$ for all k. □

Practice: Squared Hinge Penalty vs. Smoothed Hinge Penalty

Both the squared hinge penalty and the smoothed hinge penalty are smooth functions, but they have different practical implications. The squared hinge penalty typically requires a much larger penalty parameter, on the order of $\rho = O(1/\epsilon)$ as indicated by the theory, to enforce the constraints effectively. In contrast, the smoothed hinge penalty achieves similar constraint satisfaction with a significantly smaller ρ. This difference is illustrated in Figure 6.31 (right), which shows that a large penalty parameter $\rho = 800$ is needed for the squared hinge penalty, whereas the smoothed hinge penalty achieves comparable results with just $\rho = 20$. As a result, optimization of the objective function tends to be more effective when using the smoothed hinge penalty as seen in Figure 6.31 (left).

6.7.3 Constrained Learning with Fairness Constraints

Machine learning models are increasingly used in high-stakes domains such as hiring, finance, and healthcare, where biased predictions can lead to unfair outcomes for individuals from protected groups (e.g., based on race, gender, or age). Learning with fairness constraints is a framework that aims to train models that are both accurate and equitable by incorporating formal definitions of fairness directly into the training objective. Various notions of fairness have been proposed, including demographic parity, equalized odds, equal opportunity, AUC fairness, ROC fairness,

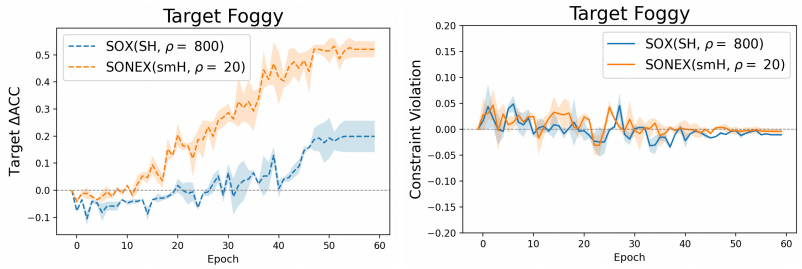


Fig. 6.31: Training curves of Target ΔACC values (left) and constraint violation (right) of different methods. The format of label is "Algorithm(penalty function, ρ)", and SH, smH mean square hinge and smoothed hinge, respectively. For more details, please refer to (Chen et al., 2025b).

and ranking fairness. Below, we present an application of constrained optimization to learning under ROC fairness constraints.

Constrained Learning with ROC Fairness

We consider a binary classification setting. Let $h(\mathbf{w};\cdot) \in \mathbb{R}$ denote a predictive model. Suppose the data are divided into two demographic groups $\mathcal{D}_p = \{(\mathbf{x}_i^p, y_i^p)\}_{i=1}^{n_p}$ and $\mathcal{D}_u = \{(\mathbf{x}_i^u, y_i^u)\}_{i=1}^{n_u}$, where $\mathbf{x}$ denotes the input data and $y \in \{1,-1\}$ denotes the class label. Traditional fairness measures usually assume the prediction is given by $\mathbb{I}(h(\mathbf{w};\mathbf{x}) > t)$ with a specific threshold. However, the threshold may be dynamically changed in practice to achieve a balance between true positive and false positive rate.

To accommodate this, a ROC fairness is introduced to ensure the ROC curves for classification of the two groups are the same, which indicates the false positive rate (FPR) and true positive rate (TPR) at all possible thresholds are equal across the two groups. Since the ROC curve is constructed with all possible thresholds, we use a set of thresholds $\Gamma = \{\tau_1, \cdots, \tau_m\}$ to define the ROC fairness. For each threshold τ, we impose a constraint that the TPR and FPR of the two groups are close, formulated as the following:

$$g_\tau^+(\mathbf{w}) = \left| \frac{1}{n_p^+} \sum_{i=1}^{n_p} \mathbb{I}(y_i^p = 1)\sigma(h(\mathbf{w};\mathbf{x}_i^p) - \tau) - \frac{1}{n_u^+} \sum_{i=1}^{n_u} \mathbb{I}(y_i^u = 1)\sigma(h(\mathbf{w};\mathbf{x}_i^u) - \tau) \right| - \kappa \le 0,$$

and

$$g_\tau^-(\mathbf{w}) = \left| \frac{1}{n_p^-} \sum_{i=1}^{n_p} \mathbb{I}(y_i^p = -1)\sigma(h(\mathbf{w};\mathbf{x}_i^p) - \tau) - \frac{1}{n_u^-} \sum_{i=1}^{n_u} \mathbb{I}(y_i^u = -1)\sigma(h(\mathbf{w};\mathbf{x}_i^u) - \tau) \right| - \kappa \le 0,$$

where $\sigma(s)$ is a surrogate of the indicator function $\mathbb{I}(s > 0)$, e.g., the sigmoid function, and $\kappa > 0$ is a tolerance parameter.

Then the learning problem can be imposed as:

$$\begin{aligned} \min_{\mathbf{w}} \quad & F(\mathbf{w}), \\ s.t. \quad & g_\tau^+(\mathbf{w}) \leq 0, g_\tau^-(\mathbf{w}) \leq 0, \forall \tau \in \Gamma. \end{aligned}$$

where $F(\mathbf{w})$ is an appropriate risk function.

By utilizing the penalty method, we solve the following problem:

$$\min_{\mathbf{w}} F(\mathbf{w}) + \frac{\rho}{2|\Gamma|} \sum_{\tau \in \Gamma} (f(g_\tau^+(\mathbf{w})) + f(g_\tau^-(\mathbf{w}))). \tag{6.101}$$

Let us define

$$\begin{aligned} g_1(\mathbf{w}; \tau) &= \frac{1}{n_p^+} \sum_{i=1}^{n_p} \mathbb{I}(y_i^p = 1) \sigma(h(\mathbf{w}; \mathbf{x}_i^p) - \tau) \\ g_2(\mathbf{w}; \tau) &= \frac{1}{n_u^+} \sum_{i=1}^{n_u} \mathbb{I}(y_i^u = 1) \sigma(h(\mathbf{w}; \mathbf{x}_i^u) - \tau). \end{aligned}$$

Since $f(\cdot)$ is a non-decreasing convex function, hence $f(|x|)$ is a convex function. Then the penalty term $f(g_\tau^+(\mathbf{w})) = f(|g_1(\mathbf{w}; \tau) - g_2(\mathbf{w}; \tau)| - \kappa)$ is a compositional of a convex function $f(\boldsymbol{g}) = f(|g_1 - g_2| - \kappa)$ and a smooth mapping $g(\mathbf{w}) = [g_1(\mathbf{w}; \tau), g_2(\mathbf{w}; \tau)]$. Hence, SONX, SONEX, ALEXR-DL can be employed to solve the above problem.

6.8 Learning Data Compositional Networks

So far, we have considered the compositional loss function, which involves comparing the output of one data $h(\mathbf{w}; \mathbf{x})$ with that of many other data. In this section, we consider compositional networks, where the computation of $h(\mathbf{w}; \mathbf{x})$ for one data $\mathbf{x}$ depends on many other data.

6.8.1 Large-scale Graph Neural Networks

Graph Neural Networks (GNNs) are a powerful class of models designed to learn representations from graph-structured data, where information is distributed across nodes and edges. Unlike traditional neural networks that operate on grid-like inputs, GNNs leverage the connectivity structure of graphs to propagate and aggregate information from a node's neighborhood, capturing both local and global patterns.

GNNs have been successfully applied to tasks such as node classification, link prediction, and graph-level classification in domains including social networks, molecular chemistry, and recommendation systems.

A key distinction in GNN-based learning lies between transductive and inductive settings. In transductive learning, the model is trained and tested on the same fixed graph, meaning all nodes (including test nodes) are present during training. Classic GNN models such as Graph Convolutional Neural (GCN) Network in this setting. In contrast, inductive methods aim to generalize to unseen nodes or entirely new graphs not available during training. GraphSAGE (Graph Sample and Aggregate) is a method that is designed for inductive learning, enabling flexible deployment in dynamic environments where new nodes or graphs continuously emerge.

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ denote a graph, where $\mathcal{V}$ is the set of nodes and $\mathcal{E}$ is the set of edges. Each node $v \in \mathcal{V}$ is associated with a feature vector $\mathbf{x}_v$. Given a node v with neighbors $\mathcal{N}(v)$, a general scheme fo updating the node's representation in layer k is following:

$$\begin{aligned}\mathbf{h}_{\mathcal{N}(v)}^{(k)} &= \text{Aggregate}\left(\left\{\mathbf{h}_u^{(k-1)} : u \in \mathcal{N}(v)\right\}\right),\\ \mathbf{h}_v^{(k)} &= \text{Update}\left(\mathbf{h}_v^{(k-1)}, \mathbf{h}_{\mathcal{N}(v)}^{(k)}\right),\end{aligned}$$

where the first step aggregates the representations of the nodes in the immediate neighborhood of node v into a single vector, and the second step updates the node's current representation $\mathbf{h}_v^{(k-1)}$, with the aggregated neighborhood vector to generate a new embedding $\mathbf{h}_v^{(k)}$.

GraphSAGE (Graph Sample and Aggregate)

GraphSAGE is a scalable inductive framework for learning node representations in large graphs. Let us consider a particular implementation of the above framework:

$$\mathcal{A}(\{\mathbf{h}_u^{(k-1)} : u \in \mathcal{N}(v) \cup \{v\}\}) = \frac{1}{|\mathcal{N}_v| + 1} \sum_{u \in \mathcal{N}(v) \cup \{v\}} \mathbf{h}_u^{(k-1)} \tag{6.102}$$

$$\mathbf{h}_v^{(k)} = \sigma\left(\mathbf{W}^{(k)} \cdot \mathcal{A}(\{\mathbf{h}_u^{(k-1)} : u \in \mathcal{N}_v \cup \{v\}\})\right), \tag{6.103}$$

where $\mathcal{A}(\cdot)$ denotes the mean operator and $\sigma(\cdot)$ is an activation function.

When working with large-scale graphs, GraphSAGE employs node sampling to ensure scalability. At each layer, a node samples a fixed number of neighbors and aggregates their features. However, as the number of layers increases, the number of nodes involved in computing a single node's embedding can grow exponentially. Specifically, if each node samples K neighbors and the model has L layers, then computing the embedding for a single node may involve up to K^L nodes. This exponential growth is known as the *neighborhood explosion problem*, which can lead to significant computational and memory overhead, especially in deep models or

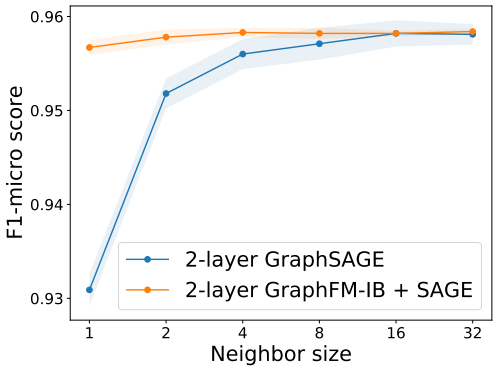


Fig. 6.32: Comparison between standard GraphSAGE and GraphSAGE with Feature Momentum on the Reddit dataset, which contains 232,965 nodes and 11,606,919 edges. Each node has an average of 49.82 neighbors. For more details, please refer to (Yu et al., 2022).

large graphs. While reducing K (e.g., to 1) can mitigate neighborhood explosion, it may also introduce high variance in the estimation of the mean operator potentially degrading model performance.

GraphSAGE with Feature Momentum

The challenge discussed earlier arises from the compositional structure of $\mathbf{h}_v^{(k)}$. To address this, we leverage a moving average estimator. Let $\mathcal{B}_v \subset \mathcal{N}(v)$ be a sub-sampled neighborhood of node v, and define $\bar{\mathcal{B}}_v = \mathcal{B}_v \cup \{v\}$. At the t-th iteration, we estimate the aggregated feature vector as follows:

$$\tilde{\mathbf{h}}_v^{(k,t)} = \begin{cases} \tilde{\mathbf{h}}_v^{(k,t-1)} & \text{if } v \notin \mathcal{D}_k, \\ (1-\gamma)\tilde{\mathbf{h}}_v^{(k,t-1)} + \gamma \hat{\mathcal{A}}\left(\left\{\hat{\mathbf{h}}_u^{(k-1,t)} : u \in \bar{\mathcal{B}}_v\right\}\right) & \text{otherwise,} \end{cases} \tag{6.104}$$

where $\mathcal{D}_k$ is the sub-sampled set of nodes updated at the k-th layer, $\gamma \in (0,1)$ is the momentum parameter, and $\hat{\mathcal{A}}(\cdot)$ is an unbiased estimator of the aggregation function $\mathcal{A}(\cdot)$ over the neighborhood $\mathcal{N}_v \cup \{v\}$. The estimator is computed as:

$$\hat{\mathcal{A}}\left(\left\{\hat{\mathbf{h}}_u^{(k-1,t)} : u \in \bar{\mathcal{B}}_v\right\}\right) = \frac{1}{|\mathcal{N}_v|+1}\hat{\mathbf{h}}_v^{(k-1,t)} + \frac{|\mathcal{N}_v|}{|\mathcal{N}_v|+1} \cdot \frac{1}{|\mathcal{B}_v|}\sum_{u \in \mathcal{B}_v} \hat{\mathbf{h}}_u^{(k-1,t)}.$$

Next, we update the feature representation at the k-th layer:

$$\hat{\mathbf{h}}_v^{(k,t)} = \sigma\left(\mathbf{W}_t^{(k)} \cdot \tilde{\mathbf{h}}_v^{(k,t)}\right). \tag{6.105}$$

This process is repeated for L layers to compute the output representation $\hat{\mathbf{h}}_v^{(L,t)}$ for sub-sampled nodes $v \in \mathcal{D}_L$, which are then used to compute the mini-batch loss. We refer to this approach as GraphSAGE with Feature Momentum.

This method effectively reduces the required number of sampled neighbors per node while maintaining the performance of using full neighborhoods; see Figure 6.32.

6.8.2 Multi-instance Learning with Attention

Multi-instance learning (MIL) refers to a setting where a bag of instances are observed for an object of interest and only one label is given to describe that object. Many real-life applications can be formulated as MIL. For example, the medical imaging data for diagnosing a patient usually consists of a series of 2D high-resolution images (e.g., CT scan), and only a single label (containing a tumor or not) is assigned to the patient.

A standard assumption for MIL is that a bag is labeled positive if at least one of its instances has a positive label, and negative if all of its instances have negative labels. The assumption implies that a MIL model must be permutation-invariant for the prediction function $h(\mathcal{X})$, where $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\}$ denotes a bag of instances. To achieve permutation invariant property, fundamental theorems of symmetric functions have been developed. In particular, a scoring function for a set of instances $\mathcal{X}$ denoted by $h(\mathcal{X}) \in \mathbb{R}$, is a symmetric function if and only if it can be decomposed as $h(\mathcal{X}) = g(\sum_{\mathbf{x}\in\mathcal{X}} \psi(\mathbf{x}))$ (Zaheer et al., 2017), where g and ψ are suitable transformations. Another theory is that a Hausdorff continuous symmetric function $h(\mathcal{X}) \in \mathbb{R}$ can be arbitrarily approximated by a function in the form $g(\max_{\mathbf{x}\in\mathcal{X}} \psi(\mathbf{x}))$ (Qi et al., 2016), where max is the element-wise vector maximum operator and ψ and g are continuous functions. These theories provide support for several widely used pooling operators used for MIL.

Deep learning with different pooling operations

Let $e(\mathbf{w}_e; \mathbf{x}) \in \mathbb{R}^{d_o}$ be the instance-level representation encoded by a neural network $\mathbf{w}_e$, $\phi(\mathbf{w}; \mathbf{x}) \in [0, 1]$ be the instance-level prediction score (after some activation function), and $h(\mathbf{w}; \mathcal{X}_i) \in [0, 1]$ be the pooled prediction score of the bag i over all its instances. Besides, $\sigma(\cdot)$ denotes the sigmoid activation.

Softmax pooling of predictions

The simplest approach is to take the maximum of predictions of all instances in the bag, i.e., $h(\mathbf{w}; \mathcal{X}) = \max_{\mathbf{x}\in\mathcal{X}} \phi(\mathbf{w}; \mathbf{x})$. However, the max operation is non-smooth, which usually causes difficulty in optimization. In practice, a smoothed-max (aka. log-sum-exp) pooling operator is used instead:

$$h(\mathbf{w}; \mathcal{X}) = \tau \log \left(\frac{1}{|\mathcal{X}|} \sum_{\mathbf{x}\in\mathcal{X}} \exp(\phi(\mathbf{w}; \mathbf{x})/\tau) \right), \tag{6.106}$$

where $\tau > 0$ is a hyperparameter and $\phi(\mathbf{w}; \mathbf{x})$ is the prediction score for instance $\mathbf{x}$.

Mean pooling of predictions

The mean pooling operator just takes the average of predictions of individual instances, i.e., $h(\mathbf{w}; \mathcal{X}) = \frac{1}{|\mathcal{X}|} \sum_{\mathbf{x} \in \mathcal{X}} \phi(\mathbf{w}; \mathbf{x})$. Indeed, smoothed-max pooling interpolates between the max pooling (with $\tau = 0$) and the mean pooling (with $\tau = \infty$).

Attention-based Pooling of features

Attention-based pooling aggregates the feature representations using attention, i.e.,

$$E(\mathbf{w}; \mathcal{X}) = \sum_{\mathbf{x} \in \mathcal{X}} \frac{\exp(g(\mathbf{w}; \mathbf{x}))}{\sum_{\mathbf{x}' \in \mathcal{X}} \exp(g(\mathbf{w}; \mathbf{x}'))} e(\mathbf{w}_e; \mathbf{x}), \tag{6.107}$$

where $g(\mathbf{w}; \mathbf{x})$ is a parametric function, e.g., $g(\mathbf{w}; \mathbf{x}) = \mathbf{w}_a^\top \tanh(Ve(\mathbf{w}_e; \mathbf{x}))$, where $V \in \mathbb{R}^{m \times d_o}$ and $\mathbf{w}_a \in \mathbb{R}^m$. Based on the aggregated feature representation, the bag level prediction can be computed by

$$h(\mathbf{w}; \mathcal{X}) = \sigma(\mathbf{w}_c^\top E(\mathbf{w}; \mathcal{X})) = \sigma\left(\sum_{\mathbf{x} \in \mathcal{X}} \frac{\exp(g(\mathbf{w}; \mathbf{x})) s(\mathbf{w}; \mathbf{x})}{\sum_{\mathbf{x}' \in \mathcal{X}} \exp(g(\mathbf{w}; \mathbf{x}'))} \right), \tag{6.108}$$

where $s(\mathbf{w}; \mathbf{x}) = \mathbf{w}_c^\top e(\mathbf{w}_e; \mathbf{x})$.

Optimization Algorithms

Given the pooled prediction $h(\mathbf{w}; \mathcal{X})$, the empirical risk minimization (ERM) problem is defined as:

$$\min_{\mathbf{w}} \frac{1}{n} \sum_{i=1}^{N} \ell_i(h(\mathbf{w}; \mathcal{X}_i)).$$

The main challenge in solving this problem lies in the computational cost of evaluating $h(\mathbf{w}; \mathcal{X}_i)$, as it involves aggregating over potentially many instances.

To address this, we employ techniques from compositional optimization. Specifically, we express the smoothed-max pooling in (6.106) as a composition $h(\mathbf{w}; \mathcal{X}_i) = f_2(f_1(\mathbf{w}; \mathcal{X}_i))$, where the functions f_1 and f_2 are defined as:

$$f_1(\mathbf{w}; \mathcal{X}_i) = \frac{1}{|\mathcal{X}_i|} \sum_{\mathbf{x}_{i,j} \in \mathcal{X}_i} \exp(\phi(\mathbf{w}; \mathbf{x}_{i,j})/\tau),$$
$$f_2(s_i) = \tau \log(s_i).$$

Similarly, we express the attention-based pooling in (6.108) as a compositional function $h(\mathbf{w}; \mathcal{X}_i) = f_2(f_1(\mathbf{w}; \mathcal{X}_i))$, with:

$$f_1(\mathbf{w}; \mathcal{X}_i) = \begin{bmatrix} \frac{1}{|\mathcal{X}_i|} \sum_{\mathbf{x}_{i,j} \in \mathcal{X}_i} \exp(g(\mathbf{w}; \mathbf{x}_{i,j})) \mathbf{w}_c^\top e(\mathbf{w}_e; \mathbf{x}_{i,j}) \\ \frac{1}{|\mathcal{X}_i|} \sum_{\mathbf{x}_{i,j} \in \mathcal{X}_i} \exp(g(\mathbf{w}; \mathbf{x}_{i,j})) \end{bmatrix}, \quad f_2(\mathbf{u}_i) = \sigma\left(\frac{[\mathbf{u}_i]_1}{[\mathbf{u}_i]_2}\right).$$

The key difference between the two pooling mechanisms is that the inner function f_1 in attention-based pooling is a vector-valued function with two components. In both cases, the computational bottleneck lies in computing $f_1(\mathbf{w}; \mathcal{X}_i)$.

To reduce this cost, we maintain a dynamic estimator $u_{i,t}$ for each bag $\mathcal{X}_i$. At iteration t, for any $\mathcal{X}_i \in \mathcal{B}_{o,t}$ (a mini-batch of bags), we update the estimator as:

$$u_{i,t} = (1-\gamma) u_{i,t-1} + \gamma f_1(\mathbf{w}_t; \mathcal{B}_{i,t}), \tag{6.109}$$

where $\mathcal{B}_{i,t} \subset \mathcal{X}_i$ is a mini-batch of instances sampled from $\mathcal{X}_i$, and $\gamma \in [0,1]$ is a smoothing parameter. For smoothed-max pooling, this becomes:

$$u_{i,t} = (1-\gamma) u_{i,t-1} + \frac{\gamma}{|\mathcal{B}_{i,t}|} \sum_{\mathbf{x}_{i,j} \in \mathcal{B}_{i,t}} \exp(\phi(\mathbf{w}_t; \mathbf{x}_{i,j})/\tau), \tag{6.110}$$

and for attention-based pooling, we update:

$$\mathbf{u}_{i,t} = (1-\gamma) \mathbf{u}_{i,t-1} + \gamma \begin{bmatrix} \frac{1}{|\mathcal{B}_{i,t}|} \sum_{\mathbf{x}_{i,j} \in \mathcal{B}_{i,t}} \exp(g(\mathbf{w}_t; \mathbf{x}_{i,j})) \delta(\mathbf{w}_t; \mathbf{x}_{i,j}) \\ \frac{1}{|\mathcal{B}_{i,t}|} \sum_{\mathbf{x}_{i,j} \in \mathcal{B}_{i,t}} \exp(g(\mathbf{w}_t; \mathbf{x}_{i,j})) \end{bmatrix}. \tag{6.111}$$

The corresponding vanilla gradient estimator for softmax pooling is:

$$\mathbf{z}_t = \frac{1}{|\mathcal{B}|} \sum_{\mathcal{X}_i \in \mathcal{B}} \ell_i'(f_2(u_{i,t})) \nabla f_2(u_{i,t}) \frac{1}{|\mathcal{B}_{i,t}|} \sum_{\mathbf{x}_{i,j} \in \mathcal{B}_{i,t}} \nabla \exp(\phi(\mathbf{w}_t; \mathbf{x}_{i,j})/\tau), \tag{6.112}$$

and for attention-based pooling:

$$\mathbf{z}_t = \tag{6.113}$$
$$\frac{1}{|\mathcal{B}|} \sum_{\mathcal{X}_i \in \mathcal{B}} \ell_i'(f_2(\mathbf{u}_{i,t})) \begin{bmatrix} \frac{1}{|\mathcal{B}_{i,t}|} \sum_{\mathbf{x}_{i,j} \in \mathcal{B}_{i,t}} \nabla \left(\exp(g(\mathbf{w}_t; \mathbf{x}_{i,j})) s(\mathbf{w}^t; \mathbf{x}_{i,j})\right) \\ \frac{1}{|\mathcal{B}_{i,t}|} \sum_{\mathbf{x}_{i,j} \in \mathcal{B}_{i,t}} \nabla \exp(g(\mathbf{w}_t; \mathbf{x}_{i,j})) \end{bmatrix}^\top \nabla f_2(\mathbf{u}_{i,t}).$$

Then we can update the model parameter $\mathbf{w}_{t+1}$ by Momentum, Adam, or Adam-W methods.

As established in Chapter 5, the theory of compositional optimization guarantees that the moving average estimators $\mathbf{u}_{i,t}$ ensure the average estimation error,

$$\frac{1}{T} \sum_{t=1}^{T} \|\mathbf{u}_{i,t} - f_1(\mathbf{w}_t; \mathcal{X}_i)\|_2^2,$$

converges to zero as $T \to \infty$, provided that the model parameters and hyperparameters are properly updated.

6.9 DRRHO Risk Minimization

As a last application of compositional optimization, we consider an emerging problems in AI. With the success of large foundation models, numerous companies and research groups have entered the race to develop state-of-the-art models. While the data and code are often proprietary, the resulting models are sometimes released publicly, such as the CLIP models from OpenAI. How can we leverage these open-weight models? We discuss three commonly used strategies and then present an emerging paradigm.

Using the Model As-Is

A straightforward strategy for leveraging open-weight foundation models is to use them as-is. This approach requires no additional training and can be deployed immediately, making it highly convenient and cost-effective. It is particularly attractive when computational resources or labeled data are limited. However, the downside is that the pretrained model may not perform well on specialized tasks or under distribution shifts, where its generic knowledge does not fully align with the requirements of the target application.

Fine-Tuning the Model

An alternative strategy is to use the pretrained model as a starting point for fine-tuning. By performing minimal task-specific training, the model can be adapted to new domains with relatively low computational and data costs. Fine-tuning generally yields better performance than using the model out-of-the-box. Nevertheless, since the model architecture remains unchanged and the updates are typically modest, the improvements in performance may be limited, particularly when the pretrained model is already near-optimal for its design.

Knowledge Distillation from the Model

A more flexible approach involves using the pretrained model as a teacher in a knowledge distillation framework. Here, a smaller or more efficient student model is trained to mimic the teacher's outputs, enabling knowledge transfer that can improve training efficiency and generalization. This strategy is particularly useful for deploying models in resource-constrained environments. The main drawback, however, is that the student model is usually less expressive than the teacher, which can cap its performance despite potential gains in speed and efficiency.

Reference Model Steering for training from scratch

An emerging learning paradigm has recently surfaced that leverages a pre-trained reference model to guide and enhance training via strategic data weighting—a process we term reference model steering. Unlike the knowledge distillation framework, reference model steering does not assume that the reference model is a stronger teacher; in fact, it can lead to the training of a model that ultimately surpasses the reference model in performance, i.e., enabling weak to strong generalization.

DRRHO Risk Minimization

Let $\mathbf{z} \sim \mathbb{P}$ denote a random data point drawn from distribution $\mathbb{P}$, and let $\mathbf{w} \in \mathcal{W}$ represent model parameters from a parameter space $\mathcal{W}$. Given a loss function $\ell(\mathbf{w}, \mathbf{z})$, the expected risk is defined as:

$$\mathcal{R}(\mathbf{w}) = \mathbb{E}_{\mathbf{z}\sim\mathbb{P}}[\ell(\mathbf{w}, \mathbf{z})].$$

Given a pretrained reference model $\mathbf{w}_{\text{ref}}$, we define a new loss $\hat{\ell}(\mathbf{w}, \cdot) = \ell(\mathbf{w}, \cdot) - \ell(\mathbf{w}_{\text{ref}}, \cdot)$, which is termed as RHO loss. Incorporating this into the distributionally robust optimization (DRO) framework (2.12), we define DRRHO risk minimization as:

$$\min_{\mathbf{w}\in\mathcal{W}} \sup_{\substack{\mathbf{p}\in\Delta \\ D_\phi(\mathbf{p}\,\|\,1/n)\le\rho/n}} \sum_{i=1}^{n} p_i \left(\ell(\mathbf{w}, \mathbf{z}_i) - \ell(\mathbf{w}_{\text{ref}}, \mathbf{z}_i)\right). \tag{6.114}$$

Theoretical guarantees for DRRHO have been developed with the χ^2 divergence, i.e., $D_\phi(\mathbf{p}\,\|\,\mathbf{q}) = \sum_{i=1}^{n} \frac{1}{2} q_i \left(\frac{p_i}{q_i} - 1\right)^2$. Under mild conditions, it can be shown that with high probability:

$$\mathcal{R}(\tilde{\mathbf{w}}_*) \le \inf_{\mathbf{w}\in\mathcal{W}} \left(\mathcal{R}(\mathbf{w}) + \sqrt{\frac{2\rho}{n} \operatorname{Var}(\ell(\mathbf{w}, \cdot) - \ell(\mathbf{w}_{\text{ref}}, \cdot))}\right) + O\left(\frac{1}{n}\right). \tag{6.115}$$

where $\tilde{\mathbf{w}}_*$ is an optimal solution to DRRHO risk minimization.

In particular, plugging in $\mathbf{w}_* = \arg\min_{\mathbf{w}\in\mathcal{W}} \mathcal{R}(\mathbf{w})$ yields:

$$\mathcal{R}(\tilde{\mathbf{w}}_*) \le \mathcal{R}(\mathbf{w}_*) + \sqrt{\frac{2\rho}{n} \operatorname{Var}(\ell(\mathbf{w}_*, \cdot) - \ell(\mathbf{w}_{\text{ref}}, \cdot))} + O\left(\frac{1}{n}\right).$$

This result provides valuable insight: if the reference model $\mathbf{w}_{\text{ref}}$ is well-trained such that $\ell(\mathbf{w}_{\text{ref}}, \cdot)$ closely matches $\ell(\mathbf{w}_*, \cdot)$ in distribution, then the variance term becomes small. As a result, DRRHO achieves better generalization than the standard $O(\sqrt{1/n})$ bound of ERM.

Furthermore, if $\mathbf{w}_{\text{ref}} \in \mathcal{W}$, we obtain a comparison in terms of excess risk:

$$\mathcal{R}(\tilde{\mathbf{w}}_*) - \mathcal{R}(\mathbf{w}_*) \le \mathcal{R}(\mathbf{w}_{\text{ref}}) - \mathcal{R}(\mathbf{w}_*) + O\left(\frac{1}{n}\right).$$

This enables a direct comparison between the DRRHO minimizer $\tilde{\mathbf{w}}_*$ and the reference model $\mathbf{w}_{\text{ref}}$ from the same hypothesis class. Suppose $\mathbf{w}_{\text{ref}}$ was trained via ERM on a dataset with m samples. Then standard generalization theory gives an excess risk of order $O(1/\sqrt{m})$. In contrast, to match this level of generalization error, DRRHO requires only $n = O(\sqrt{m})$ samples—significantly improving over the $O(m)$ sample complexity required by ERM without a reference model.

Optimization Algorithms

When the CVaR is used defined by $\phi(t) = 1$ if $t \leq n/k$ and $\phi(t) = \infty$ otherwise, the DRRHO risk reduces to the average of the top-k RHO losses:

$$\min_{\mathbf{w}} F(\mathbf{w}) := \frac{1}{k} \sum_{i=1}^{k} \left(\ell(\mathbf{w}, \mathbf{z}_{[i]}) - \ell(\mathbf{w}_{\text{ref}}, \mathbf{z}_{[i]}) \right), \tag{6.116}$$

where $\mathbf{z}_{[i]}$ denotes the data point ranked i-th in descending order based on its RHO loss. This problem can be equivalently reformulated as:

$$\min_{\mathbf{w}, \nu} \frac{1}{k} \sum_{i=1}^{n} [\ell(\mathbf{w}, \mathbf{z}_i) - \ell(\mathbf{w}_{\text{ref}}, \mathbf{z}_i) - \nu]_+ + \nu, \tag{6.117}$$

which is more amenable to gradient-based optimization techniques.

When DRRHO risk is defined using KL divergence regularization, the objective becomes:

$$\min_{\mathbf{w}} \tau \log \left(\frac{1}{n} \sum_{i=1}^{n} \exp \left(\frac{\ell(\mathbf{w}, \mathbf{z}_i) - \ell(\mathbf{w}_{\text{ref}}, \mathbf{z}_i)}{\tau} \right) \right). \tag{6.118}$$

This formulation can be optimized by simply replacing the loss in Algorithm 24 with the RHO loss. The vanilla gradient at iteration t is estimated by:

$$\frac{1}{B} \sum_{i \in \mathcal{B}_t} \frac{\exp\left(\frac{\ell(\mathbf{w}_t, \mathbf{z}_i) - \ell(\mathbf{w}_{\text{ref}}, \mathbf{z}_i)}{\tau} \right)}{u_t} \nabla \ell(\mathbf{w}_t, \mathbf{z}_i), \tag{6.119}$$

where u_t is the MA estimator of the inner function value. This gradient estimator naturally assigns higher weights to data points with larger RHO losses, thereby prioritizing samples with high learnability during training.

Finally, when DRRHO is formulated with a KL-divergence constraint, the optimization problem becomes:

$$\min_{\mathbf{w}} \min_{\tau \geq 0} \tau \log \left(\frac{1}{n} \sum_{i=1}^{n} \exp \left(\frac{\ell(\mathbf{w}, \mathbf{z}_i) - \ell(\mathbf{w}_{\text{ref}}, \mathbf{z}_i)}{\tau} \right) \right) + \frac{\tau \rho}{n}. \tag{6.120}$$

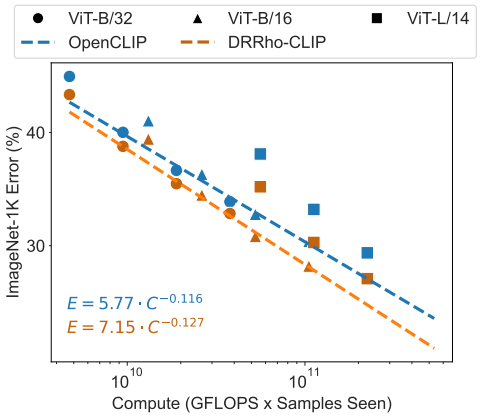


Fig. 6.33: Scaling performance of OpenCLIP and DRRho-CLIP, which uses the OpenAI CLIP model as the reference model. We conduct experiments of the two methods under different settings to fit scaling laws, as shown in the bottom left corner. For more details, please refer to (Wei et al., 2025).

This formulation can be optimized using techniques similar to those introduced in the first section of this chapter.

DRRHO-CLIP with a Reference Model

We now consider applying the DRRHO risk minimization framework to CLIP. Given the established connection between robust global contrastive loss and DRO, as shown in (6.44) and (6.45), it is straightforward to incorporate the RHO loss into the training objective. Define the following loss components:

$$\begin{aligned}
\ell_1(\mathbf{w}; \mathbf{x}_i, \mathbf{t}_i, \mathbf{t}) &= s(\mathbf{w}; \mathbf{x}_i, \mathbf{t}) - s(\mathbf{w}; \mathbf{x}_i, \mathbf{t}_i), \\
\ell_2(\mathbf{w}; \mathbf{x}_i, \mathbf{t}_i, \mathbf{x}) &= s(\mathbf{w}; \mathbf{x}, \mathbf{t}_i) - s(\mathbf{w}; \mathbf{x}_i, \mathbf{t}_i), \\
\ell_1(\mathbf{w}_{\text{ref}}; \mathbf{x}_i, \mathbf{t}_i, \mathbf{t}) &= s(\mathbf{w}_{\text{ref}}; \mathbf{x}_i, \mathbf{t}) - s(\mathbf{w}_{\text{ref}}; \mathbf{x}_i, \mathbf{t}_i), \\
\ell_2(\mathbf{w}_{\text{ref}}; \mathbf{x}_i, \mathbf{t}_i, \mathbf{x}) &= s(\mathbf{w}_{\text{ref}}; \mathbf{x}, \mathbf{t}_i) - s(\mathbf{w}_{\text{ref}}; \mathbf{x}_i, \mathbf{t}_i),
\end{aligned}$$

where $s(\cdot;\cdot,\cdot)$ denotes the similarity function, and $\mathbf{w}_{\text{ref}}$ is a pretrained reference model.

Using these definitions, we modify the original objective in (6.49) to incorporate the RHO loss:

$$\begin{aligned}
\min_{\mathbf{w},\tau_1,\tau_2} \frac{1}{n}\sum_{i=1}^{n} \tau_1 \log\left(\frac{1}{|\mathcal{T}_i^-|}\sum_{\mathbf{t}\in\mathcal{T}_i^-} \exp\left(\frac{\ell_1(\mathbf{w}; \mathbf{x}_i, \mathbf{t}_i, \mathbf{t}) - \ell_1(\mathbf{w}_{\text{ref}}; \mathbf{x}_i, \mathbf{t}_i, \mathbf{t})}{\tau_1}\right)\right) + \tau_1\rho \\
+ \frac{1}{n}\sum_{i=1}^{n} \tau_2 \log\left(\frac{1}{|\mathcal{I}_i^-|}\sum_{\mathbf{x}\in\mathcal{I}_i^-} \exp\left(\frac{\ell_2(\mathbf{w}; \mathbf{x}_i, \mathbf{t}_i, \mathbf{x}) - \ell_2(\mathbf{w}_{\text{ref}}; \mathbf{x}_i, \mathbf{t}_i, \mathbf{x})}{\tau_2}\right)\right) + \tau_2\rho.
\end{aligned} \tag{6.121}$$

This objective can be optimized using an algorithm similar to that used in the CLIP training. Empirical results show that this approach significantly reduces sample

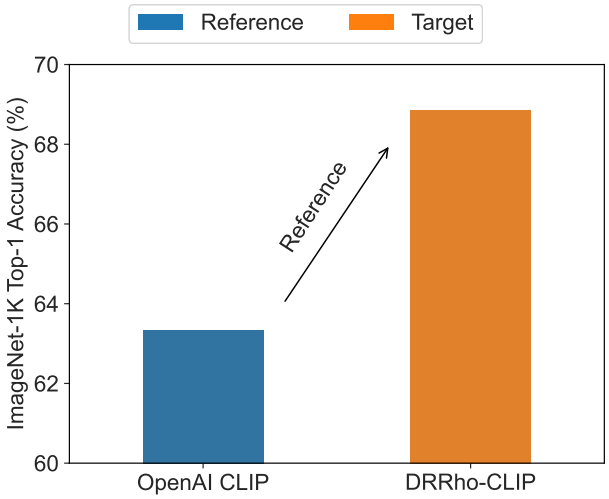


Fig. 6.34: Comparison between a target model (ViT-B/16) trained by DRRRHO-CLIP and the reference model it leverages. OpenAI CLIP (ViT-B/32) was trained on a private 400M dataset with 12.8B samples seen and 32768 batch size. DRRho-CLIP model was trained on DFN-192M with 1.28B samples seen and 5120 batch size, and using OpenAI CLIP as a reference model. DRRHO-CLIP training took 376 GPU hours on 8 H100 (2 days), OpenAI CLIP (ViT-L/14) model was trained on 256 V100 with 12 days, which gives an estimate of 256*12*24/11.6=6356 GPU hours for training ViT-B/32 as its FLOPs is 11.6 smaller. For more details, please refer to (Wei et al., 2025).

complexity and improves the empirical scaling law (see Figure 6.33), while also achieving weak to strong generalization (see Figure 6.34).

6.10 History and Notes

DRO and GDRO.

We first formulated KL-regularized Distributionally Robust Optimization (DRO) as a stochastic compositional optimization (SCO) problem in (Qi et al., 2021b), utilizing STORM-based estimators. This line of research was further developed in (Qi et al., 2020), which introduced an attentional biased stochastic momentum method for KL-regularized DRO with specific applications in imbalanced data classification. Subsequently, we extended both the algorithmic framework and theoretical analysis to address KL-constrained DRO (Qi et al., 2023). Collectively, these works demon-

strate the advantages of employing compositional optimization techniques over traditional primal-dual methods for solving DRO problems.

The formulation of FCCO for group DRO (GDRO) was initially identified in (Hu et al., 2024b). Building on this, Wang and Yang (2023) applied the ALEXR algorithm to convex group DRO, demonstrating significant improvements over traditional stochastic primal-dual methods. Most recently, the application of SONEX to non-convex group DRO within the context of deep learning was investigated by Chen et al. (2025b).

Stochastic AUC and NDCG Optimization.

Stochastic AUC maximization has a long-standing history in machine learning, as detailed in our survey (Yang and Ying, 2023). The formulation of AUC maximization with a square surrogate loss as a minimax optimization problem was first introduced by Ying et al. (2016b). Building on this foundation, we developed the first convergence analysis for stochastic non-convex minimax optimization in the context of deep AUC maximization (Liu et al., 2020). While this work was inspired by our previous work on weakly-convex strongly-concave minimax optimization (Rafique et al., 2022), it established a superior complexity bound by leveraging the PL condition. These theoretical results were subsequently strengthened in (Guo et al., 2023).

This line of research eventually facilitated our winning entry in the CheXpert competition for X-ray image classification (Yuan et al., 2021), which also introduced the AUC-margin minimax objective. Notably, all of these proposed methods utilize a double-loop algorithmic structure. The single-loop PDMA and PDAdam methods for deep AUC maximization was first proposed and analyzed in our work (Guo et al., 2021b). The compositional training method for deep AUC maximization that facilitates the feature learning and classifier learning in a unified framework was proposed in our work (Yuan et al., 2022b).

The SOAP algorithm represents the first method of its kind to offer a convergence guarantee that does not rely on the use of large batch sizes, which has challenged the computer vision and machine learning communities for many years (see references in (Qi et al., 2021c)). The SOPA and SOPAs algorithms for one-way partial AUC maximization and STOAs for two-way partial AUC maximization were developed and analyzed in (Zhu et al., 2022b). The STACO algorithm for two-way partial AUC maximization was proposed in (Zhou et al., 2025). These stdudies have addressed long-standing open problems for efficient partial AUC maximization with convergence guarantee (Kar et al., 2014; Narasimhan and Agarwal, 2013).

The formulation of stochastic NDCG optimization as FCCO was proposed in our work (Qiu et al., 2022), which also developed a multi-block bilevel optimization formulation and algorithm for optimizing top-K NDCG. The complexity for multi-block bilevel optimization was improved in (Hu et al., 2023) by using the MSVR estimators.

The design and benchmark of LibAUC library was presented in (Yuan et al., 2023a).

Discriminative Learning of Foundation models.

The SogCLR algorithm was inspired by the SOX framework for FCCO; its advantages over SimCLR, particularly regarding efficiency with small batch sizes in unimodal contrastive learning, were demonstrated in (Yuan et al., 2022c). Building on this, we introduced iSogCLR in (Qiu et al., 2023) to optimize individualized temperatures. This advancement was also informed by our previous research on KL-constrained DRO (Qi et al., 2023).

Subsequently, we proposed TempNet (Qiu et al., 2024), which has been successfully applied to CLIP training and the pretraining of large language models (LLMs). Furthermore, a comprehensive evaluation of FCCO-based techniques for distributed CLIP training was recently provided in (Wei et al., 2024).

The discriminative fine-tuning approach of LLMs was proposed in our work (Guo et al., 2025). The DisCO method for fine-tuning large reasoning models was developed in our work (Li et al., 2025).

FCCO for Constrained Learning.

The application of compositional optimization techniques to penalty methods for constrained learning dates back to (Ermoliev and Wets, 1988). The first non-asymptotic analysis of the penalty method with a squared hinge penalty function for non-convex inequality constrained optimization based on FCCO was conducted in our work (Li et al., 2024). This work investigated the problem within the context of continual learning under zero-forgetting constraints and established a complexity of $O(1/\epsilon^7)$ for finding an ϵ-KKT solution. Additionally, we developed a theoretical framework to characterize the benefits of network expansion in facilitating constrained learning with non-forgetting constraints. The ROC fairness constraint was first considered in (Vogel et al., 2020).

Subsequent advancements have further improved the complexity of penalty based methods based on FCCO. By employing SONX for the hinge penalty, the complexity was reduced to $O(1/\epsilon^6)$ (Yang et al., 2025). More recently, the introduction of SONEX and a double-loop ALEXR method for the squared hinge penalty achieved a complexity of $O(1/\epsilon^5)$ (Chen et al., 2025b). This currently represents the state-of-the-art complexity for penalty methods in non-convex constrained optimization.

Learning with data compositional networks.

Graph convolutional neural network was proposed by Kipf and Welling (2017). GraphSAGE was developed in (Hamilton et al., 2017). The use of compositional optimization techniques, specifically incorporating feature momentum for large-scale Graph Neural Network (GNN) learning, was introduced in our previous work (Yu et al., 2022). Furthermore, the application of compositional optimization to multi-instance learning, utilizing compositional pooling operations, was first proposed

in (Zhu et al., 2023a). Attention-based pooling for multi-instance learning was proposed by Ilse et al. (2018).

DRRHO risk minimization.

The development of DRRHO risk minimization framework and its application to CLIP training was introduced in our work (Wei et al., 2025). The theoretical analysis of this method is largely built upon the foundations of DRO (Namkoong and Duchi, 2017), while the conceptual idea of using the RHO loss for data selection in a mini-batch was originally proposed in (Mindermann et al., 2022).

Chapter 7
Afterword

Dear Readers:

Congratulations on making it this far in the book. Even if you haven't read every chapter in full, I hope you've found parts of it useful and inspiring. If you are a practitioner, I hope this book convince you that theory can, at times, be genuinely useful in practice.

Before concluding this book, I would like to reflect on my journey into compositional optimization for advanced machine learning, which began in 2019. Before that, I was primarily focused on traditional stochastic optimization theory. During that time, we developed a stochastic algorithm for solving non-convex minimax optimization problems. In 2019, I spent a year in industry, where conversations with young professionals made me realize the importance of practicability. After combing back from the leave, I started to think about how to make the theory more practical. The first project was to apply our non-convex minimax optimization to AUC maximization for learning deep neural networks, leading us to achieve first place in the Stanford CheXpert competition for classifying X-ray images organized by Andrew Ng's ML group in 2020. In late 2020, my friend Shuiwang Ji at Texas A&M University introduced me to the MIT AICures challenge, which aimed to identify few molecules with properties suitable for COVID-19 drug development among many. Motivated by this challenge, I formulated the optimization problem of maximizing the empirical estimator of areas under precision-recall curves, known as average precision. This led me to define the novel finite-sum coupled compositional optimization (FCCO) framework. We developed the first algorithm for FCCO in 2021, which ultimately helped us win the MIT AICures Challenge.

As I explored further, I discovered broad applications of this framework in ML and AI, specifically in addressing the computational challenges inherent in contrastive learning, learning to rank, discriminative learning and continual learning. This series of work eventually led to the development of the LibAUC library for empirical X-risk minimization, which has since been downloaded over 100,000 times by researchers and developers across more than 85 countries. It also helped us to develop CLIP models better than OpenAI's models with 15 times less compute budget.

As I reflect on the journey that led to this book, I am reminded of the principle of '知行合一' (Zhi Xing He Yi) by Wang, Yangming (a legendary sage of ancient China), cited in the preface. It is often translated as 'unity of knowledge and action', which I interpret as the idea that theory should guide practice, and practice can, in turn, inspire theory.

For decades, the field of machine learning has largely been framed through the lens of Empirical Risk Minimization (ERM). This book argues that such a view is increasingly insufficient for modern AI systems. As we have seen throughout these chapters, the "X" in EXM represents a diverse class of often non-decomposable objectives, such as AUC, ranking measures, cross-entropy loss with expensive normalization, and contrastive losses, which define the frontier of modern AI. The development of the LibAUC library and the success of the EXM framework in AI challenges have shown us that when we move beyond standard stochastic optimization, we unlock new levels of performance in critical domains like medical imaging and drug discovery. Yet, despite these successes, the systematic study of EXM is just beginning.

During my years as a graduate student, I immersed myself in many books on optimization and machine learning, which were instrumental in shaping my mathematical foundation. Now, after more than a decade of study and research, I am humbled to synthesize these insights—together with my own findings—into this book. I hope that the methods and theories presented here are not viewed as a final destination, but rather as a starting point. My hope is that this work encourages researchers to look beyond standard training loops and to explore new forms of "X-risks" that better capture the complexity of modern learning systems and the nuances of human intelligence and societal needs. Ultimately, I look forward to seeing how the next generation of researchers will build upon these ideas to bridge elegant mathematical theory with transformative real-world applications.

Finally, I should note that this book may contain typographical errors and may inevitably omit some important related works. I would be deeply grateful to readers who are willing to share corrections, suggestions, or feedback to help improve future revisions.

References

Alacaoglu A, Cevher V, Wright SJ (2025) On the complexity of a simple primal-dual coordinate method. Mathematical Programming

Amari S (1967) A theory of adaptive pattern classifier. IEEE Transactions on Electronic Computers EC-16(3):279–307

An X, Zhu X, Gao Y, Xiao Y, Zhao Y, Feng Z, Wu L, Qin B, Zhang M, Zhang D, Fu Y (2021) Partial fc: Training 10 million identities on a single machine. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops, pp 1445–1449

Arjevani Y, Carmon Y, Duchi JC, Foster DJ, Srebro N, Woodworth B (2022) Lower bounds for non-convex stochastic optimization. Math Program 199(1–2):165–214, DOI 10.1007/s10107-022-01822-7, URL https://doi.org/10.1007/s10107-022-01822-7

Ben-Tal A, Teboulle M (1986a) Expected utility, penalty functions, and duality in stochastic nonlinear programming. Management Science 32(11):1445–1466

Ben-Tal A, Teboulle M (1986b) Expected utility, penalty functions, and duality in stochastic nonlinear programming. Management Science 32(11):1445–1466, DOI 10.1287/mnsc.32.11.1445, URL https://doi.org/10.1287/mnsc.32.11.1445

Ben-Tal A, Teboulle M (2007) An old-new concept of convex risk measures: the optimized certainty equivalent. Mathematical Finance 17(3):449–476

Ben-Tal A, El Ghaoui L, Nemirovski A (2009a) Robust Optimization. Princeton Series in Applied Mathematics, Princeton University Press

Ben-Tal A, Ghaoui LE, Nemirovski A (2009b) Robust Optimization. Princeton Series in Applied Mathematics, Princeton University Press, Princeton, NJ

Ben-Tal A, den Hertog D, De Waegenaere A, Melenberg B, Rennen G (2013) Robust solutions of optimization problems affected by uncertain probabilities. Management Science 59(2):341–357, DOI 10.1287/mnsc.1120.1641, URL https://doi.org/10.1287/mnsc.1120.1641

Bertsekas D (2005) Control of uncertain systems with a set-membership description of the uncertainty

Bertsekas DP (2009) Convex Optimization Theory. Athena Scientific

Bishop CM (2006) Pattern Recognition and Machine Learning (Information Science and Statistics). Springer-Verlag, Berlin, Heidelberg

Bommasani R, Hudson DA, Adeli E, Altman RB, Arora S, von Arx S, Bernstein MS, Bohg J, Bosselut A, Brunskill E, Brynjolfsson E, Buch S, Card D, Castellon R, Chatterji NS, Chen AS, Creel K, Davis JQ, Demszky D, Donahue C, Doumbouya M, Durmus E, Ermon S, Etchemendy J, Ethayarajh K, Fei-Fei L, Finn C, Gale T, Gillespie LE, Goel K, Goodman ND, Grossman S, Guha N, Hashimoto T, Henderson P, Hewitt J, Ho DE, Hong J, Hsu K, Huang J, Icard T, Jain S, Jurafsky D, Kalluri P, Karamcheti S, Keeling G, Khani F, Khattab O, Koh PW, Krass MS, Krishna R, Kuditipudi R, et al (2021) On the opportunities and risks of foundation models. CoRR abs/2108.07258, URL https://arxiv.org/abs/2108.07258, 2108.07258

Boyd K, Eng KH, Page CD (2013) Area under the precision-recall curve: Point estimates and confidence intervals. In: Machine Learning and Knowledge Discovery in Databases. ECML PKDD 2013, Springer, pp 451–466, DOI 10.1007/978-3-642-40994-3_29

Boyd S, Vandenberghe L (2004) Convex Optimization. Cambridge University Press

Bracken J, McGill JT (1973) Mathematical programs with optimization problems in the constraints. Operations Research 21:37–44

Brown DB (2007) Large deviations bounds for estimating conditional value-at-risk. Oper Res Lett 35(6):722–730, DOI 10.1016/j.orl.2007.01.001, URL https://doi.org/10.1016/j.orl.2007.01.001

Calafiore GC (2007) Ambiguous risk measures and optimal robust portfolios. SIAM Journal on Optimization 18(3):853–877, DOI 10.1137/050639379, URL https://doi.org/10.1137/050639379

Cao K, Wei C, Gaidon A, Arechiga N, Ma T (2019) Learning imbalanced datasets with label-distribution-aware margin loss. In: Advances in Neural Information Processing Systems (NeurIPS), vol 32, pp 1567–1578

Cao Z, Qin T, Liu TY, Tsai MF, Li H (2007) Learning to rank: From pairwise approach to listwise approach. In: Proceedings of the 24th International Conference on Machine Learning, pp 129–136

Cauchy AL (1847) Méthode générale pour la résolution des systèmes d'équations simultanées. Comptes Rendus de l'Académie des Sciences 25:536, reprinted in *Œuvres complètes*, Série 1, Tome 10, pp. 399–402. Gallica digital document.

Chang KW, Hsieh CJ, Lin CJ (2008) Coordinate descent method for large-scale l2-loss linear support vector machines. Journal of Machine Learning Research 9(45):1369–1398, URL http://jmlr.org/papers/v9/chang08a.html

Chen L, Ma Y, Zhang J (2025a) Near-optimal nonconvex-strongly-convex bilevel optimization with fully first-order oracles. Journal of Machine Learning Research 26(109):1–56, URL http://jmlr.org/papers/v26/23-1104.html

Chen T, Sun Y, Yin W (2021a) Closing the gap: Tighter analysis of alternating stochastic gradient methods for bilevel problems. In: Beygelzimer A, Dauphin Y, Liang P, Vaughan JW (eds) Advances in Neural Information Processing Systems, URL https://openreview.net/forum?id=r6cNUjS8cm0

Chen T, Sun Y, Yin W (2021b) Solving stochastic compositional optimization is nearly as easy as solving stochastic optimization. IEEE Transactions on Signal Processing 69:4937–4948, DOI 10.1109/TSP.2021.3092377

Chen X, Wang B, Yang M, Lin Q, Yang T (2025b) Stochastic momentum methods for non-smooth non-convex finite-sum coupled compositional optimization. In: The Thirty-ninth Annual Conference on Neural Information Processing Systems, URL https://openreview.net/forum?id=kSgZiAAwDU

Chiang CK, Yang T, Lee CJ, Mahdavi M, Lu CJ, Jin R, Zhu S (2012) Online optimization with gradual variations. In: Mannor S, Srebro N, Williamson RC (eds) Proceedings of the 25th Annual Conference on Learning Theory, PMLR, Edinburgh, Scotland, Proceedings of Machine Learning Research, vol 23, pp 6.1–6.20, URL https://proceedings.mlr.press/v23/chiang12.html

Chung KL (1954) On a Stochastic Approximation Method. The Annals of Mathematical Statistics 25(3):463 – 483, DOI 10.1214/aoms/1177728716, URL https://doi.org/10.1214/aoms/1177728716

Cortes C, Mohri M (2003) AUC optimization vs. error rate minimization. Advances in Neural Information Processing Systems 16, URL https://proceedings.neurips.cc/paper_files/paper/2003/file/2518-auc-optimization-vs-error-rate-minimization.pdf

Crammer K, Singer Y (2002) On the algorithmic implementation of multiclass kernel-based vector machines. J Mach Learn Res 2:265–292

Csiszár I (1967) Information-type measures of difference of probability distributions and indirect observations. Studia Scientiarum Mathematicarum Hungarica 2:299–318

Cutkosky A, Orabona F (2019) Momentum-based variance reduction in non-convex SGD, Curran Associates Inc., Red Hook, NY, USA

Dang CD, Lan G (2015) Stochastic block mirror descent methods for nonsmooth and stochastic optimization. SIAM Journal on Optimization 25(2):856–882

Danskin J (1967) The Theory of Max-min and Its Applications to Weapons Allocation Problems. Econometrics and operations research, Springer, URL https://books.google.ca/books?id=bvrfAQAACAAJ

Daskalakis C, Ilyas A, Syrgkanis V, Zeng H (2018) Training GANs with optimism. In: International Conference on Learning Representations, URL https://openreview.net/forum?id=SJJySbbAZ

Daubechies I, Defrise M, Mol CD (2004) An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. Communications on Pure and Applied Mathematics 57(11):1413–1457, DOI 10.1002/cpa.20042

Davis D, Drusvyatskiy D (2019) Stochastic model-based minimization of weakly convex functions. SIAM Journal on Optimization 29(1):207–239, DOI 10.1137/18M1178244

Dekel O, Singer Y (2005) Data-driven online to batch conversions. In: Weiss Y, Schölkopf B, Platt J (eds) Advances in Neural Information Processing Systems, MIT Press, vol 18, URL https://proceedings.neurips.cc/paper_files/paper/2005/file/4a5876b450b45371f6cfe5047ac8cd45-Paper.pdf

Delage E, Ye Y (2010) Distributionally robust optimization under moment uncertainty with application to data-driven problems. Operations Research 58(3):595–612

Deleu T, Bengio Y (2021) Structured sparsity inducing adaptive optimizers for deep learning. ArXiv abs/2102.03869, URL https://api.semanticscholar.org/CorpusID:231846689

Dodd LE, Pepe MS (2003) Partial AUC estimation and regression. Biometrics 59(3):614–623, DOI 10.1111/1541-0420.00071, URL https://doi.org/10.1111/1541-0420.00071

Drusvyatskiy D, Paquette C (2019) Efficiency of minimizing compositions of convex functions and smooth maps. Mathematical Programming 178:503–558

Drusvyatskiy D, Ioffe AD, Lewis AS (2021) Nonsmooth optimization using taylor-like models: error bounds, convergence, and termination criteria. Mathematical Programming 185:357–383

Duchi J, Singer Y (2009) Efficient online and batch learning using forward backward splitting. J Mach Learn Res 10:2899–2934

Duchi J, Hazan E, Singer Y (2011) Adaptive subgradient methods for online learning and stochastic optimization. Journal of Machine Learning Research 12:2121–2159

Duchi JC, Ruan F (2017) Stochastic methods for composite and weakly convex optimization problems. arXiv preprint arXiv:170308570 URL https://arxiv.org/abs/1703.08570, 1703.08570

Duchi JC, Ruan F (2018) Stochastic methods for composite and weakly convex optimization problems. SIAM Journal on Optimization 28(4):3229–3259, DOI 10.1137/17M1135086

Duchi JC, Glynn PW, Namkoong H (2022) Statistics of robust optimization: A generalized empirical likelihood approach. Mathematics of Operations Research 47(2):882–910, DOI 10.1287/moor.2020.1085, URL https://doi.org/10.1287/moor.2020.1085

Dupačová J (1966) On minimax solutions of stochastic linear programming problems. Časopis pro pěstování matematiky 091(4):423–430, URL http://eudml.org/doc/20949

Ermoliev Y, Wets RJB (eds) (1988) Numerical Techniques for Stochastic Optimization, Springer Series in Computational Mathematics, vol 10. Springer-Verlag

Ermoliev YM (1976) Methods of Stochastic Programming. Monographs in Optimization and Operations Research, Nauka, Moscow

Fan J, Li R (2001) Variable selection via nonconcave penalized likelihood and its oracle properties. Journal of the American Statistical Association 96(456):1348–1360, DOI 10.1198/016214501753382273

Fang C, Li CJ, Lin Z, Zhang T (2018) Spider: Near-optimal non-convex optimization via stochastic path-integrated differential estimator. In: Bengio S, Wallach H, Larochelle H, Grauman K, Cesa-Bianchi N, Garnett R (eds) Advances in Neural Information Processing Systems, Curran Associates, Inc., vol 31, URL https://proceedings.neurips.cc/paper_files/paper/2018/file/1543843a4723ed2ab08e18053ae6dc5b-Paper.pdf

Fazel M, Hindi H, Boyd SP (2001) A rank minimization heuristic with application to minimum order system approximation. In: 2001 IEEE International Conference on Control Applications, IEEE, pp 1347–1352

Fletcher R (1982) A model algorithm for composite nondifferentiable optimization problems. Mathematical Programming Study 17:67–76

Fletcher R, Watson GA (1980) First and second order conditions for a class of non-differentiable optimization problems. Mathematical Programming 18:291–307

Frankel SP (1950) Convergence rates of iterative treatments of partial differential equations. Mathematics of Computation 4:65–75, URL https://api.semanticscholar.org/CorpusID:119690385

Freund Y, Schapire RE (1997) A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. J Comput Syst Sci 55(1):119–139, DOI 10.1006/jcss.1997.1504

Freund Y, Iyer R, Schapire RE, Singer Y (2003) An efficient boosting algorithm for combining preferences. J Mach Learn Res 4(null):933–969

Ghadimi S, Lan G (2012) Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization i: A generic algorithmic framework. SIAM Journal on Optimization 22(4):1469–1492, DOI 10.1137/110848864

Ghadimi S, Lan G (2013) Stochastic first- and zeroth-order methods for nonconvex stochastic programming. SIAM Journal on Optimization 23(4):2341–2368, DOI 10.1137/120880811

Ghadimi S, Wang M (2018) Approximation methods for bilevel programming. URL https://arxiv.org/abs/1802.02246, 1802.02246

Ghadimi S, Ruszczyński A, Wang M (2020) A single timescale stochastic approximation method for nested stochastic optimization. SIAM Journal on Optimization 30(1):960–979, DOI 10.1137/18M1230542

Ghosh A, Kumar H, Sastry PS (2017) Robust loss functions under label noise for deep neural networks. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol 31

Goldstein AA (1964) Convex programming in hilbert space. Bulletin of the American Mathematical Society 70(5):709–710

Gong P, Zhang C, Lu Z, Huang J, Ye J (2013) A general iterative shrinkage and thresholding algorithm for non-convex regularized optimization problems. In: Dasgupta S, McAllester D (eds) Proceedings of the 30th International Conference on Machine Learning, PMLR, Atlanta, Georgia, USA, Proceedings of Machine Learning Research, vol 28, pp 37–45, URL https://proceedings.mlr.press/v28/gong13a.html

Green DM, Swets JA (1966) Signal Detection Theory and Psychophysics. John Wiley and Sons Inc., New York, NY

Guo S, Hong I, Balmaseda V, Yu C, Qiu L, Liu X, Jiang H, Zhao T, Yang T (2025) Discriminative finetuning of generative large language models without reward models and human preference data. In: Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025, OpenReview.net, URL https://openreview.net/forum?id=1jutKQ5R8T

Guo Z, Hu Q, Zhang L, Yang T (2021a) Randomized stochastic variance-reduced methods for stochastic bilevel optimization. CoRR abs/2105.02266, URL https://arxiv.org/abs/2105.02266, 2105.02266

Guo Z, Xu Y, Yin W, Jin R, Yang T (2021b) Unified convergence analysis for adaptive optimization with moving average estimator. Mach Learn 114(4), DOI 10.1007/s10994-024-06650-8, URL https://doi.org/10.1007/s10994-024-06650-8

Guo Z, Yan Y, Yuan Z, Yang T (2023) Fast objective & duality gap convergence for non-convex strongly-concave min-max problems with PL condition. J Mach Learn Res 24:148:1–148:63, URL https://jmlr.org/papers/v24/21-1471.html

Hadsell R, Chopra S, LeCun Y (2006) Dimensionality reduction by learning an invariant mapping. In: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), IEEE, vol 2, pp 1735–1742, DOI 10.1109/CVPR.2006.100

Hamilton WL, Ying R, Leskovec J (2017) Inductive representation learning on large graphs. In: Proceedings of the 31st International Conference on Neural Information Processing Systems, Curran Associates Inc., Red Hook, NY, USA, NIPS'17, p 1025–1035

Hanley JA, McNeil BJ (1982) The meaning and use of the area under a receiver operating characteristic (roc) curve. Radiology 143(1):29–36

Hardt M, Recht B, Singer Y (2016) Train faster, generalize better: Stability of stochastic gradient descent. In: Proceedings of the 33rd International Conference on Machine Learning (ICML), PMLR, pp 1225–1234

Hazan E, Kale S (2011) Beyond the regret minimization barrier: an optimal algorithm for stochastic strongly-convex optimization. In: Kakade SM, von Luxburg U (eds) Proceedings of the 24th Annual Conference on Learning Theory, PMLR, Budapest, Hungary, Proceedings of Machine Learning Research, vol 19, pp 421–436, URL https://proceedings.mlr.press/v19/hazan11a.html

Hazan E, Agarwal A, Kale S (2007) Logarithmic regret algorithms for online convex optimization. Mach Learn 69(2–3):169–192, DOI 10.1007/s10994-007-5016-8, URL https://doi.org/10.1007/s10994-007-5016-8

Hinton G (2018) Neural networks for machine learning, lecture 6. Coursera online course

Hong M, Wai HT, Wang Z, Yang Z (2020) A two-timescale stochastic algorithm framework for bilevel optimization: Complexity analysis and application to actor-critic. SIAM Journal on Optimization 33(1):147–180

Hsieh CJ, Chang KW, Lin CJ, Keerthi SS, Sundararajan S (2008) A dual coordinate descent method for large-scale linear svm. In: Proceedings of the 25th International Conference on Machine Learning, Association for Computing Machinery, New York, NY, USA, ICML '08, p 408–415, DOI 10.1145/1390156.1390208, URL https://doi.org/10.1145/1390156.1390208

Hu Q, Qiu Z, Guo Z, Zhang L, Yang T (2023) Blockwise stochastic variance-reduced methods with parallel speedup for multi-block bilevel optimization. CoRR abs/2305.18730, DOI 10.48550/ARXIV.2305.18730, URL https://doi.org/10.48550/arXiv.2305.18730, 2305.18730

Hu Q, Qi Q, Lu Z, Yang T (2024a) Single-loop stochastic algorithms for difference of max-structured weakly convex functions. In: The Thirty-eighth Annual Conference on Neural Information Processing Systems, URL https://openreview.net/forum?id=NhtBXSNXKA

Hu Q, Qi Q, Lu Z, Yang T (2024b) Single-loop stochastic algorithms for difference of max-structured weakly convex functions. In: Globersons A, Mackey L, Belgrave D, Fan A, Paquet U, Tomczak JM, Zhang C (eds) Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 -

15, 2024, URL http://papers.nips.cc/paper_files/paper/2024/hash/67e79c8e9b11f068a7cafd79505175c0-Abstract-Conference.html

Hu W, Niu G, Sato I, Sugiyama M (2018) Does distributionally robust supervised learning give robust classifiers? In: Dy J, Krause A (eds) Proceedings of the 35th International Conference on Machine Learning, PMLR, Proceedings of Machine Learning Research, vol 80, pp 2029–2037, URL https://proceedings.mlr.press/v80/hu18a.html

Hu W, Li CJ, Lian X, Liu J, Yuan H (2019) Efficient smooth non-convex stochastic compositional optimization via stochastic recursive gradient descent. In: Wallach H, Larochelle H, Beygelzimer A, d'Alché-Buc F, Fox E, Garnett R (eds) Advances in Neural Information Processing Systems, Curran Associates, Inc., vol 32, URL https://proceedings.neurips.cc/paper_files/paper/2019/file/21ce689121e39821d07d04faab328370-Paper.pdf

Hu Y, Zhang S, Chen X, He N (2020) Biased stochastic first-order methods for conditional stochastic optimization and applications in meta learning. In: Proceedings of the 34th International Conference on Neural Information Processing Systems, Curran Associates Inc., Red Hook, NY, USA, NIPS '20

Huang F, Gao S, Pei J, Huang H (2022) Accelerated zeroth-order and first-order momentum methods from mini to minimax optimization. J Mach Learn Res 23(1)

Huo Z, Gu B, Liu J, Huang H (2018) Accelerated method for stochastic composition optimization with nonsmooth regularization. In: Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence, AAAI Press, AAAI'18/IAAI'18/EAAI'18

Ilharco G, Wortsman M, Wightman R, Gordon C, Carlini N, Taori R, Dave A, Shankar V, Namkoong H, Miller J, Hajishirzi H, Farhadi A, Schmidt L (2021) Openclip. DOI 10.5281/zenodo.5143773, URL https://doi.org/10.5281/zenodo.5143773, if you use this software, please cite it as below.

Ilse M, Tomczak J, Welling M (2018) Attention-based deep multiple instance learning. In: Dy J, Krause A (eds) Proceedings of the 35th International Conference on Machine Learning, PMLR, Proceedings of Machine Learning Research, vol 80, pp 2127–2136, URL https://proceedings.mlr.press/v80/ilse18a.html

Iouditski A, Nesterov Y (2010) Primal-dual subgradient methods for minimizing uniformly convex functions. arXiv: Optimization and Control URL https://api.semanticscholar.org/CorpusID:117741989

Järvelin K, Kekäläinen J (2000) Ir evaluation methods for retrieving highly relevant documents. In: Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Association for Computing Machinery, New York, NY, USA, SIGIR '00, p 41–48, DOI 10.1145/345508.345545, URL https://doi.org/10.1145/345508.345545

Ji K, Yang J, Liang Y (2020) Bilevel optimization: Convergence analysis and enhanced design. In: International Conference on Machine Learning, URL https://api.semanticscholar.org/CorpusID:235825903

Jiang W, Li G, Wang Y, Zhang L, Yang T (2022) Multi-block-single-probe variance reduced estimator for coupled compositional optimization. In: Koyejo S, Mohamed S, Agarwal A, Belgrave D, Cho K, Oh A (eds) Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022, URL http://papers.nips.cc/paper_files/paper/2022/hash/d13ee73683fd5567e5c07634a25cd7b8-Abstract-Conference.html

Jiang W, Qin J, Wu L, Chen C, Yang T, Zhang L (2023) Learning unnormalized statistical models via compositional optimization. In: Krause A, Brunskill E, Cho K, Engelhardt B, Sabato S, Scarlett J (eds) International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA, PMLR, Proceedings of Machine Learning Research, vol 202, pp 15105–15124, URL https://proceedings.mlr.press/v202/jiang23g.html

Jin L, Ma J, Liu Z, Gromov A, Defazio A, Xiao L (2025) PARQ: Piecewise-affine regularized quantization. In: Forty-second International Conference on Machine Learning, URL https://openreview.net/forum?id=8PCxOlwbIn

Johnson R, Zhang T (2013) Accelerating stochastic gradient descent using predictive variance reduction. In: Burges C, Bottou L, Welling M, Ghahramani Z, Weinberger K (eds) Advances in Neural Information Processing Systems, Curran Associates, Inc., vol 26, URL https://proceedings.neurips.cc/paper_files/paper/2013/file/ac1dd209cbcc5e5d1c6e28598e8cbbe8-Paper.pdf

Jordan K, Jin Y, Boza V, Jiacheng Y, Cesista F, Newhouse L, Bernstein J (2024) Muon: An optimizer for hidden layers in neural networks. URL https://kellerjordan.github.io/posts/muon/

Juditsky A, Nemirovski A, Tauvel C (2011) Solving variational inequalities with stochastic mirror-prox algorithm. Stochastic Systems 1(1):17–58

Kar P, Narasimhan H, Jain P (2014) Online and stochastic gradient methods for non-decomposable loss functions. In: Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1, MIT Press, Cambridge, MA, USA, NIPS'14, p 694–702

Karush W (1939) Minima of functions of several variables with inequalities as side constraints. M.sc. thesis, University of Chicago, Chicago, Illinois

Khanduri P, Zeng S, Hong M, Wai HT, Wang Z, Yang Z (2021) A near-optimal algorithm for stochastic bilevel optimization via double-momentum. In: Proceedings of the 35th International Conference on Neural Information Processing Systems, Curran Associates Inc., Red Hook, NY, USA, NIPS '21

Kiefer J, Wolfowitz J (1952) Stochastic Estimation of the Maximum of a Regression Function. The Annals of Mathematical Statistics 23(3):462 – 466, DOI 10.1214/aoms/1177729392, URL https://doi.org/10.1214/aoms/1177729392

Kingma DP, Ba J (2014) Adam: A method for stochastic optimization. CoRR abs/1412.6980, URL https://api.semanticscholar.org/CorpusID:6628106

Kingma DP, Ba J (2015) Adam: A method for stochastic optimization. In: Bengio Y, LeCun Y (eds) ICLR (Poster), URL http://dblp.uni-trier.de/db/conf/iclr/iclr2015.html#KingmaB14

Kipf TN, Welling M (2017) Semi-supervised classification with graph convolutional networks. In: International Conference on Learning Representations, URL https://openreview.net/forum?id=SJU4ayYgl

Kivinen J, Warmuth MK (1997) Exponentiated gradient versus gradient descent for linear predictors. Inf Comput 132(1):1–63, DOI 10.1006/inco.1996.2612, URL https://doi.org/10.1006/inco.1996.2612

Koltchinskii V (2011) Oracle Inequalities in Empirical Risk Minimization and Sparse Recovery Problems. Ecole d' Eté de Probabilités de Saint-Flour XXXVIII-2008, Springer

Korpelevich GM (1976) The extragradient method for finding saddle points and other problems. URL https://api.semanticscholar.org/CorpusID:118602977

Kouvelis P, Yu G (1997) Robust Discrete Optimization and Its Applications, 1st edn. Springer, Boston, MA, DOI 10.1007/978-1-4757-2620-6

Kuhn HW, Tucker AW (2014) Nonlinear Programming, Springer Basel, Basel, pp 247–258. DOI 10.1007/978-3-0348-0439-4_11, URL https://doi.org/10.1007/978-3-0348-0439-4_11

Kwon J, Kwon D, Wright S, Nowak R (2023) A fully first-order method for stochastic bilevel optimization. In: Proceedings of the 40th International Conference on Machine Learning, JMLR.org, ICML'23

Lacoste-Julien S, Schmidt M, Bach FR (2012) A simpler approach to obtaining an o(1/t) convergence rate for the projected stochastic subgradient method. CoRR abs/1212.2002, URL http://arxiv.org/abs/1212.2002, 1212.2002

Lan G (2012) An optimal method for stochastic composite optimization. Math Program 133(1–2):365–397, DOI 10.1007/s10107-010-0434-y, URL https://doi.org/10.1007/s10107-010-0434-y

Lan G (2020) First-order and Stochastic Optimization Methods for Machine Learning, 1st edn. Springer Series in the Data Sciences, Springer International Publishing, Cham

Lan G, Ouyang Y, Zhang Z (2023) Optimal and parameter-free gradient minimization methods for convex and nonconvex optimization. URL https://api.semanticscholar.org/CorpusID:265506741

Lapin M, Hein M, Schiele B (2018) Analysis and optimization of loss functions for multiclass, top-k, and multilabel classification. IEEE Transactions on Pattern Analysis and Machine Intelligence 40(7):1533–1554, DOI 10.1109/TPAMI.2017.2751607, URL https://doi.org/10.1109/TPAMI.2017.2751607

Lee J, Park S, Shin J (2020) Learning bounds for risk-sensitive learning. ArXiv abs/2006.08138, URL https://api.semanticscholar.org/CorpusID:219686788

Lei YX, Ying Y (2019) Fine-grained analysis of stability and generalization for stochastic gradient descent. In: International Conference on Machine Learning (ICML), pp 5809–5819

Lewis A, Wright S (2009) A proximal method for composite minimization. Mathematical Programming 158, DOI 10.1007/s10107-015-0943-9

Li G, Yu W, Yao Y, Tong W, Liang Y, Lin Q, Yang T (2024) Model developmental safety: A safety-centric method and applications in vision-language models. CoRR abs/2410.03955, DOI 10.48550/ARXIV.2410.03955, URL https://doi.org/10.48550/arXiv.2410.03955, 2410.03955

Li G, Lin M, Galanti T, Tu Z, Yang T (2025) DisCO: Reinforcing large reasoning models with discriminative constrained optimization. In: The Thirty-ninth Annual Conference on Neural Information Processing Systems, URL https://openreview.net/forum?id=zzUXS4f91r

Lin T, Jin C, Jordan M (2020) On gradient descent ascent for nonconvex-concave minimax problems. In: III HD, Singh A (eds) Proceedings of the 37th International Conference on Machine Learning, PMLR, Proceedings of Machine Learning Research, vol 119, pp 6083–6093, URL https://proceedings.mlr.press/v119/lin20a.html

Lions PL, Mercier B (1979) Splitting algorithms for the sum of two nonlinear operators. SIAM Journal on Numerical Analysis 16(6):964–979, DOI 10.1137/0716071

Littlestone N (1988) Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. Mach Learn 2(4):285–318, DOI 10.1023/A:1022869011914, URL https://doi.org/10.1023/A:1022869011914

Littlestone N, Warmuth MK (1994) The weighted majority algorithm. Information and Computation 108(2):212–261

Liu B, Ye M, Wright S, Stone P, Liu Q (2022) Bome! bilevel optimization made easy: a simple first-order approach. In: Proceedings of the 36th International Conference on Neural Information Processing Systems, Curran Associates Inc., Red Hook, NY, USA, NIPS '22

Liu M, Yuan Z, Ying Y, Yang T (2020) Stochastic AUC maximization with deep neural networks. In: 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020, OpenReview.net, URL https://openreview.net/forum?id=HJepXaVYDr

Liu R, Liu X, Zeng S, Zhang J, Zhang Y (2021) Value-function-based sequential minimization for bi-level optimization. CoRR abs/2110.04974, URL https://arxiv.org/abs/2110.04974, 2110.04974

Luenberger D (1973) Introduction to Linear and Nonlinear Programming. Addison-Wesley Publishing Company, URL https://books.google.com/books?id=1aCrPQAACAAJ

Luo ZQ, Tseng P (1992) On the convergence of the coordinate descent method for convex differentiable minimization. J Optim Theory Appl 72(1):7–35, DOI 10.1007/BF00939948, URL https://doi.org/10.1007/BF00939948

Ma J, Xiao L (2025) Quantization through piecewise-affine regularization: Optimization and statistical guarantees. URL https://arxiv.org/abs/2508.11112, 2508.11112

Mahdavi M, Jin R (2013) Mixedgrad: An $o(1/t)$ convergence rate algorithm for stochastic smooth optimization. In: Advances in Neural Information Processing Systems, vol 26, URL https://proceedings.neurips.cc/paper/2013/file/f73b76ce8949fe29bf2a537cfa420e8f-Paper.pdf

Marcum JI (1947) A statistical theory of target detection by pulsed radar. Tech. Rep. RM-754, RAND Corporation, Santa Monica, CA, URL https://www.rand.org/pubs/research_memoranda/RM754.html

Martinet B (1972) Détermination approchée d'un point fixe d'une application pseudo-contractante. cas de l'application prox. Comptes Rendus de l'Académie des Sciences, Paris, Série A 274:163–165

Mindermann S, Brauner JM, Razzak MT, Sharma M, Kirsch A, Xu W, Höltgen B, Gomez AN, Morisot A, Farquhar S, Gal Y (2022) Prioritized training on points that are learnable, worth learning, and not yet learnt. In: Chaudhuri K, Jegelka S, Song L, Szepesvari C, Niu G, Sabato S (eds) Proceedings of the 39th International Conference on Machine Learning, PMLR, Proceedings of Machine Learning Research, vol 162, pp 15630–15649, URL https://proceedings.mlr.press/v162/mindermann22a.html

Mohri M, Rostamizadeh A, Talwalkar A (2018) Foundations of Machine Learning, 2nd edn. MIT Press

Morgan W, Greiff W, Henderson J (2004) Direct maximization of average precision by hill-climbing, with a comparison to a maximum entropy approach. In: Proceedings of HLT-NAACL 2004: Short Papers, Association for Computational Linguistics, USA, HLT-NAACL-Short '04, p 93–96

Namkoong H, Duchi JC (2017) Variance-based regularization with convex objectives. In: Proceedings of the 31st International Conference on Neural Information Processing Systems, Curran Associates Inc., Red Hook, NY, USA, NIPS'17, p 2975–2984

Narasimhan H, Agarwal S (2013) A structural SVM based approach for optimizing partial auc. In: Dasgupta S, McAllester D (eds) Proceedings of the 30th International Conference on Machine Learning, PMLR, Atlanta, Georgia, USA, Proceedings of Machine Learning Research, vol 28, pp 516–524, URL https://proceedings.mlr.press/v28/narasimhan13.html

Nemirovski A (2004) Prox-method with rate of convergence o(1/t) for variational inequalities with lipschitz continuous monotone operators and smooth convex-concave saddle point problems. SIAM Journal on Optimization 15(1):229–251, DOI 10.1137/S1052623403425629

Nemirovski A, Yudin D (1978) On cezari's convergence of the steepest descent method for approximating saddle point of convex-concave functions. Soviet Mathematics Doklady 19:341–362

Nemirovski A, Juditsky A, Lan G, Shapiro A (2009) Robust stochastic approximation approach to stochastic programming. SIAM Journal on Optimization 19(4):1574–1609, DOI 10.1137/070704277

Nemirovski AS, Yudin DB (1977) Information complexity of strongly convex optimization. Ekonomika i Matematicheskie Metody 13(3):550–559, translated into English in *MATEKON*

Nemirovsky AS, Yudin DB (1983) Problem Complexity and Method Efficiency in Optimization, Wiley-Interscience Series in Discrete Mathematics, vol 15. John Wiley and Sons, New York

Nesterov Y (1983) A method for solving the convex programming problem with convergence rate $o(1/k^2)$. Proceedings of the USSR Academy of Sciences 269:543–547, URL https://api.semanticscholar.org/CorpusID:145918791

Nesterov Y (2004) Introductory Lectures on Convex Programming: A Basic Course. Springer

Nesterov Y (2012) Efficiency of coordinate descent methods on huge-scale optimization problems. SIAM Journal on Optimization 22(2):341–362, DOI 10.1137/100802001

Nguyen LM, Liu J, Scheinberg K, Takávc M (2017) Sarah: a novel method for machine learning problems using stochastic recursive gradient. In: Proceedings of the 34th International Conference on Machine Learning - Volume 70, JMLR.org, ICML'17, p 2613–2621

Orabona F (2019) A modern introduction to online learning. CoRR abs/1912.13213, URL http://arxiv.org/abs/1912.13213, 1912.13213

Ortega JM, Rheinboldt WC (1970) Iterative solution of nonlinear equations in several variables. Academic Press, New York

Pazy GB (1979) Ergodic convergence to a zero of the sum of monotone operators in hilbert space. Journal of Mathematical Analysis and Applications 72:383–390

Polyak B (1964) Some methods of speeding up the convergence of iteration methods. USSR Computational Mathematics and Mathematical Physics 4(5):1–17, URL https://www.sciencedirect.com/science/article/pii/0041555364901375

Polyak BT, Juditsky AB (1992) Acceleration of Stochastic Approximation by Averaging. SIAM Journal on Control and Optimization 30(4):838–855

Qi C, Su H, Mo K, Guibas LJ (2016) Pointnet: Deep learning on point sets for 3d classification and segmentation. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp 77–85, URL https://api.semanticscholar.org/CorpusID:5115938

Qi Q, Xu Y, Jin R, Yin W, Yang T (2020) Attentional-biased stochastic gradient descent. Trans Mach Learn Res 2023, URL https://api.semanticscholar.org/CorpusID:255125618

Qi Q, Guo Z, Xu Y, Jin R, Yang T (2021a) An online method for a class of distributionally robust optimization with non-convex objectives. In: Proceedings of the 35th International Conference on Neural Information Processing Systems, Curran Associates Inc., Red Hook, NY, USA, NIPS '21

Qi Q, Guo Z, Xu Y, Jin R, Yang T (2021b) An online method for A class of distributionally robust optimization with non-convex objectives. In: Ranzato M, Beygelzimer A, Dauphin YN, Liang P, Vaughan JW (eds) Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual, pp 10067–10080

Qi Q, Luo Y, Xu Z, Ji S, Yang T (2021c) Stochastic optimization of areas under precision-recall curves with provable convergence. In: Proceedings of the 35th International Conference on Neural Information Processing Systems, Curran Associates Inc., Red Hook, NY, USA, NIPS '21

Qi Q, Lyu J, Chan K, Bai E, Yang T (2023) Stochastic constrained DRO with a complexity independent of sample size. Trans Mach Learn Res 2023, URL https://openreview.net/forum?id=VpaXrBFYZ9

Qiu S, Yang Z, Wei X, Ye J, Wang Z (2020) Single-timescale stochastic nonconvex-concave optimization for smooth nonlinear td learning. ArXiv abs/2008.10103, URL https://api.semanticscholar.org/CorpusID:221266692

Qiu Z, Hu Q, Zhong Y, Zhang L, Yang T (2022) Large-scale stochastic optimization of NDCG surrogates for deep learning with provable convergence. In: Chaudhuri K, Jegelka S, Song L, Szepesvári C, Niu G, Sabato S (eds) International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA, PMLR, Proceedings of Machine Learning Research, vol 162, pp 18122–18152, URL https://proceedings.mlr.press/v162/qiu22a.html

Qiu Z, Hu Q, Yuan Z, Zhou D, Zhang L, Yang T (2023) Not all semantics are created equal: Contrastive self-supervised learning with automatic temperature individualization. In: Krause A, Brunskill E, Cho K, Engelhardt B, Sabato S, Scarlett J (eds) International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA, PMLR, Proceedings of Machine Learning Research, vol 202, pp 28389–28421, URL https://proceedings.mlr.press/v202/qiu23a.html

Qiu Z, Guo S, Xu M, Zhao T, Zhang L, Yang T (2024) To cool or not to cool? temperature network meets large foundation models via DRO. In: Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024, OpenReview.net, URL https://openreview.net/forum?id=YWuSLBkfOw

Rafique H, Liu M, Lin Q, Yang T (2018) Weakly-convex−concave min−max optimization: provable algorithms and applications in machine learning. Optimization Methods and Software 37:1087 – 1121, URL https://api.semanticscholar.org/CorpusID:233790522

Rafique H, Liu M, Lin Q, Yang T (2022) Weakly-convex-concave min-max optimization: provable algorithms and applications in machine learning. Optim Methods Softw 37(3):1087–1121, DOI 10.1080/10556788.2021.1895152, URL https://doi.org/10.1080/10556788.2021.1895152

Rakhlin A, Sridharan K (2013) Optimization, learning, and games with predictable sequences. In: Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2, Curran Associates Inc., Red Hook, NY, USA, NIPS'13, p 3066–3074

Rakhlin A, Shamir O, Sridharan K (2012) Making gradient descent optimal for strongly convex stochastic optimization. In: Proceedings of the 29th International Coference on International Conference on Machine Learning, Omnipress, Madison, WI, USA, ICML'12, p 1571–1578

Recht B, Wright SJ (2025) Optimization for Modern Data Analysis. Cambridge University Press, this is a hypothetical example; details may vary for actual publications.

Robbins H, Monro S (1951) A stochastic approximation method. Annals of Mathematical Statistics 22:400–407

Rockafellar RT (1970a) Convex Analysis. Princeton University Press

Rockafellar RT (1970b) Convex analysis. Princeton Mathematical Series, Princeton University Press, Princeton, N. J.

Rockafellar RT (1976) Monotone operators and the proximal point algorithm. SIAM Journal on Control and Optimization 14(5):877–898, DOI 10.1137/0314056

Rosenblatt F (1962) Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms. Spartan Books, Washington, D.C.

Rustem B, Howe M (2002) Algorithms for Worst-Case Design and Applications to Risk Management. Princeton University Press, Princeton, NJ

Sagawa S, Koh PW, Hashimoto TB, Liang P (2019) Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. CoRR abs/1911.08731, URL http://arxiv.org/abs/1911.08731, 1911.08731

Scarf H (1958) A min-max solution of an inventory problem. In: Studies in the Mathematical Theory of Inventory and Production, Stanford University Press, pp 201–209

Schmidt M, Le Roux N, Bach F (2017) Minimizing finite sums with the stochastic average gradient. Math Program 162(1 – 2):83 – 112, DOI 10.1007/s10107-016-1030-6, URL https://doi.org/10.1007/s10107-016-1030-6

Shalev-Shwartz S, Ben-David S (2014) Understanding Machine Learning: From Theory to Algorithms. Cambridge University Press

Shalev-Shwartz S, Wexler Y (2016) Minimizing the maximal loss: How and why? CoRR abs/1602.01690, URL http://arxiv.org/abs/1602.01690, 1602.01690

Shalev-Shwartz S, Singer A, Srebro N (2007) Pegasos: Primal estimated subgradient solver for svm. In: Proceedings of the 24th International Conference on Machine Learning, pp 807–814

Shamir O, Zhang T (2013) Stochastic gradient descent for non-smooth optimization: Convergence results and optimal averaging schemes. In: Dasgupta S, McAllester D (eds) Proceedings of the 30th International Conference on Machine Learning, PMLR, Atlanta, Georgia, USA, Proceedings of Machine Learning Research, vol 28, pp 71–79, URL https://proceedings.mlr.press/v28/shamir13.html

Shapiro A, Kleywegt AJ (2002) Minimax analysis of stochastic problems. Optimization Methods and Software 17(3):523–542

Shen H, Chen T (2023) On penalty-based bilevel gradient descent method. In: Krause A, Brunskill E, Cho K, Engelhardt B, Sabato S, Scarlett J (eds) Proceedings of the 40th International Conference on Machine Learning, PMLR, Proceedings of Machine Learning Research, vol 202, pp 30992–31015, URL https://proceedings.mlr.press/v202/shen23c.html

Sohn K (2016) Improved deep metric learning with multi-class n-pair loss objective. In: Proceedings of the 30th International Conference on Neural Information Processing Systems, Curran Associates Inc., Red Hook, NY, USA, NIPS'16, p 1857–1865

Spackman KA (1989) Signal detection theory: valuable tools for evaluating inductive learning. In: Proceedings of the Sixth International Workshop on Machine Learning, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, p 160–163

Tibshirani R (1996) Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society (Series B) 58:267–288

Tieleman T, Hinton G (2012) Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural networks for machine learning 4(2):26–31

Tseng P (1990) Dual ascent methods for problems with strictly convex costs and linear constraints: A unified approach. SIAM Journal on Control and Optimization 28(1):214–242

Tseng P (2001) Convergence of a block coordinate descent method for nondifferentiable minimization. J Optim Theory Appl 109(3):475–494, DOI 10.1023/A:1017501703105, URL https://doi.org/10.1023/A:1017501703105

Tseng P, Bertsekas DP (1987) Relaxation methods for problems with strictly convex separable costs and linear constraints. Math Program 38(3):303–321

Verrelst H, Moreau Y, Vandewalle J, Timmerman D (1998) Use of a multi-layer perceptron to predict malignancy in ovarian tumors. In: Proceedings of the 1997 Conference on Advances in Neural Information Processing Systems 10, MIT Press, Cambridge, MA, USA, NIPS '97, p 978–984

Vogel R, Bellet A, Cl'emenccon S (2020) Learning fair scoring functions: Bipartite ranking under roc-based fairness constraints. In: International Conference on Artificial Intelligence and Statistics, URL https://api.semanticscholar.org/CorpusID:224899598

Wang B, Yang T (2022) Finite-sum coupled compositional stochastic optimization: Theory and applications. In: Chaudhuri K, Jegelka S, Song L, Szepesvári C, Niu G, Sabato S (eds) International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA, PMLR, Proceedings of Machine Learning Research, vol 162, pp 23292–23317, URL https://proceedings.mlr.press/v162/wang22ak.html

Wang B, Yang T (2023) A near-optimal single-loop stochastic algorithm for convex finite-sum coupled compositional optimization. In: International Conference on Machine Learning, URL https://api.semanticscholar.org/CorpusID:265658854

Wang B, Lei Y, Ying Y, Yang T (2025) On discriminative probabilistic modeling for self-supervised representation learning. In: The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025, OpenReview.net, URL https://openreview.net/forum?id=s15HrqCqbr

Wang G, Yang M, Zhang L, Yang T (2022) Momentum accelerates the convergence of stochastic AUPRC maximization. In: Camps-Valls G, Ruiz FJR, Valera I (eds) International Conference on Artificial Intelligence and Statistics, AISTATS 2022, 28-30 March 2022, Virtual Event, PMLR, Proceedings of Machine Learning Research, vol 151, pp 3753–3771, URL https://proceedings.mlr.press/v151/wang22b.html

Wang M, Fang EX, Liu H (2017a) Stochastic compositional gradient descent: algorithms for minimizing compositions of expected-value functions. Math Program 161(1–2):419–449, DOI 10.1007/s10107-016-1017-3, URL https://doi.org/10.1007/s10107-016-1017-3

Wang M, Liu J, Fang EX (2017b) Accelerating stochastic composition optimization. Journal of Machine Learning Research 18(105):1–23, URL http://jmlr.org/papers/v18/16-504.html

Warga J (1963) Minimizing certain convex functions. Journal of the Society for Industrial and Applied Mathematics 11(3):588–593

Wei X, Ye F, Yonay O, Chen X, Sun B, Tao D, Yang T (2024) Fastclip: A suite of optimization techniques to accelerate CLIP training with limited resources. CoRR abs/2407.01445, DOI 10.48550/ARXIV.2407.01445, URL https://doi.org/10.48550/arXiv.2407.01445, 2407.01445

Wei X, Lin MC, Ye F, Song F, Cao L, Thai MT, Yang T (2025) Model steering: Learning with a reference model improves generalization bounds and scaling laws. In: Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025, OpenReview.net, URL https://openreview.net/forum?id=QC4dfobOLQ

Wei X, Zhou L, Wang B, Lin CJ, Yang T (2026) A geometry-aware efficient algorithm for compositional entropic risk minimization. arXiv

Weinberger KQ, Saul LK (2009) Distance metric learning for large margin nearest neighbor classification. Journal of Machine Learning Research 10(9):207–244, URL http://jmlr.org/papers/v10/weinberger09a.html

Widrow B, Hoff ME (1960) Adaptive switching circuits. IRE WESCON Convention Record 4:96–104

Xu Y, Lin Q, Yang T (2017) Stochastic convex optimization: Faster local growth implies faster global convergence. In: Precup D, Teh YW (eds) Proceedings of the 34th International Conference on Machine Learning, PMLR, Proceedings of Machine Learning Research, vol 70, pp 3821–3830, URL https://proceedings.mlr.press/v70/xu17a.html

Xu Y, Jin R, Yang T (2019a) Non-asymptotic analysis of stochastic methods for non-smooth non-convex regularized problems. In: Wallach H, Larochelle H, Beygelzimer A, d'Alché-Buc F, Fox E, Garnett R (eds) Advances in Neural Information Processing Systems, Curran Associates, Inc., vol 32

Xu Y, Zhu S, Yang S, Zhang C, Jin R, Yang T (2019b) Learning with non-convex truncated losses by SGD. In: Globerson A, Silva R (eds) Proceedings of the Thirty-Fifth Conference on Uncertainty in Artificial Intelligence, UAI 2019, Tel Aviv, Israel, July 22-25, 2019, AUAI Press, Proceedings of Machine Learning Research, vol 115, pp 701–711, URL http://proceedings.mlr.press/v115/xu20b.html

Yan L, Dodier R, Mozer MC, Wolniewicz R (2003) Optimizing classifier performance via an approximation to the wilcoxon-mann-whitney statistic. In: Proceedings of the 20th International Conference on Machine Learning (ICML), AAAI Press, pp 848–855

Yan Y, Xu Y, Lin Q, Liu W, Yang T (2020a) Optimal epoch stochastic gradient descent ascent methods for min-max optimization. arXiv: Optimization and Control URL https://api.semanticscholar.org/CorpusID:226148475

Yan Y, Xu Y, Lin Q, Liu W, Yang T (2020b) Optimal epoch stochastic gradient descent ascent methods for min-max optimization. In: Larochelle H, Ranzato M, Hadsell R, Balcan M, Lin H (eds) Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual, URL https://proceedings.neurips.cc/paper/2020/hash/3f8b2a81da929223ae025fcec26dde0d-Abstract.html

Yang F, Koyejo S (2020) On the consistency of top-k surrogate losses. In: International Conference on Machine Learning (ICML), PMLR, pp 10727–10735

Yang H, Lu K, Lyu X, Hu F (2019) Two-way partial AUC and its properties. Statistical Methods in Medical Research 28(1):184–195, DOI 10.1177/0962280217718866, URL https://doi.org/10.1177/0962280217718866

Yang J, Ji K, Liang Y (2021) Provably faster algorithms for bilevel optimization. In: Proceedings of the 35th International Conference on Neural Information Processing Systems, Curran Associates Inc., Red Hook, NY, USA, NIPS '21

Yang L, Jin R (2006) Distance metric learning: A comprehensive survey. Tech. Rep. 2, Department of Computer Science and Engineering, Michigan State University

Yang M, Li G, Hu Q, Lin Q, Yang T (2025) Single-loop algorithms for stochastic non-convex optimization with weakly-convex constraints. CoRR abs/2504.15243, DOI 10.48550/ARXIV.2504.15243, URL https://doi.org/10.48550/arXiv.2504.15243, 2504.15243

Yang T (2022) Algorithmic foundation of empirical x-risk minimization. arXiv preprint arXiv:220600439

Yang T, Ying Y (2022) Auc maximization in the era of big data and ai: A survey. ACM Comput Surv 55(8), DOI 10.1145/3554729, URL https://doi.org/10.1145/3554729

Yang T, Ying Y (2023) AUC maximization in the era of big data and AI: A survey. ACM Comput Surv 55(8):172:1–172:37, DOI 10.1145/3554729, URL https://doi.org/10.1145/3554729

Yang T, Lin Q, Li Z (2016) Unified convergence analysis of stochastic momentum methods for convex and non-convex optimization. URL https://arxiv.org/abs/1604.03257, 1604.03257

Ye J, Zhu D, Zhu Q (1997) Exact penalization and necessary optimality conditions for generalized bilevel programming problems. SIAM Journal on Optimization 7(2):481–507

Ying Y, Wen L, Lyu S (2016a) Stochastic online AUC maximization. In: Advances in Neural Information Processing Systems, Curran Associates, Inc., vol 29, pp 451–459

Ying Y, Wen L, Lyu S (2016b) Stochastic online auc maximization. In: Proceedings of the 30th International Conference on Neural Information Processing Systems, Curran Associates Inc., Red Hook, NY, USA, NIPS'16, p 451–459

Yu H, Wang L, Wang B, Liu M, Yang T, Ji S (2022) Graphfm: Improving large-scale GNN training via feature momentum. In: Chaudhuri K, Jegelka S, Song L, Szepesvári C, Niu G, Sabato S (eds) International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA, PMLR, Proceedings of Machine Learning Research, vol 162, pp 25684–25701, URL https://proceedings.mlr.press/v162/yu22g.html

Yuan M, Lin Y (2006) Model selection and estimation in regression with grouped variables. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 68(1):49–67

Yuan Z, Yan Y, Sonka M, Yang T (2021) Large-scale robust deep AUC maximization: A new surrogate loss and empirical studies on medical image classification. In: 2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021, IEEE, pp 3020–3029, DOI 10.1109/ICCV48922.2021.00303, URL https://doi.org/10.1109/ICCV48922.2021.00303

Yuan Z, Guo Z, Chawla N, Yang T (2022a) Compositional training for end-to-end deep AUC maximization. In: International Conference on Learning Representations, URL https://openreview.net/forum?id=gPvB4pdu_Z

Yuan Z, Guo Z, Chawla NV, Yang T (2022b) Compositional training for end-to-end deep AUC maximization. In: The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022, OpenReview.net, URL https://openreview.net/forum?id=gPvB4pdu_Z

Yuan Z, Wu Y, Qiu Z, Du X, Zhang L, Zhou D, Yang T (2022c) Provable stochastic optimization for global contrastive learning: Small batch does not harm performance. In: Chaudhuri K, Jegelka S, Song L, Szepesvári C, Niu G, Sabato S (eds) International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA, PMLR, Proceedings of Machine Learning Research, vol 162, pp 25760–25782, URL https://proceedings.mlr.press/v162/yuan22b.html

Yuan Z, Zhu D, Qiu Z, Li G, Wang X, Yang T (2023a) Libauc: A deep learning library for x-risk optimization. In: Singh AK, Sun Y, Akoglu L, Gunopulos D, Yan X, Kumar R, Ozcan F, Ye J (eds) Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2023, Long Beach, CA, USA, August 6-10, 2023, ACM, pp 5487–5499, DOI 10.1145/3580305.3599861, URL https://doi.org/10.1145/3580305.3599861

Yuan Z, Zhu D, Qiu ZH, Li G, Wang X, Yang T (2023b) Libauc: A deep learning library for x-risk optimization. In: 29th SIGKDD Conference on Knowledge Discovery and Data Mining

Zaheer M, Kottur S, Ravanbhakhsh S, Póczos B, Salakhutdinov R, Smola AJ (2017) Deep sets. In: Proceedings of the 31st International Conference on Neural Information Processing Systems, Curran Associates Inc., Red Hook, NY, USA, NIPS'17, p 3394–3404

Zhang CH (2010) Nearly unbiased variable selection under minimax concave penalty. The Annals of Statistics 38(2):894–942

Zhang J, Xiao L (2019) A stochastic composite gradient method with incremental variance reduction. In: Wallach H, Larochelle H, Beygelzimer A, d'Alché-Buc F, Fox E, Garnett R (eds) Advances in Neural Information Processing Systems, Curran Associates, Inc., vol 32, URL https://proceedings.neurips.cc/paper_files/paper/2019/file/a68259547f3d25ab3c0a5c0adb4e3498-Paper.pdf

Zhang L, Mahdavi M, Jin R (2013) Linear convergence with condition number independent access of full gradients. In: Burges C, Bottou L, Welling M, Ghahramani Z, Weinberger K (eds) Advances in Neural Information Processing Systems, Curran Associates, Inc., vol 26, URL https://proceedings.neurips.cc/paper_files/paper/2013/file/37f0e884fbad9667e38940169d0a3c95-Paper.pdf

Zhang T (2004a) Solving large scale linear prediction problems using stochastic gradient descent algorithms. In: Greiner R, Schuurmans D (eds) Proceedings of the Twenty-First International Conference on Machine Learning, ICML 2004, ACM, pp 919–926

Zhang T (2004b) Statistical analysis of some multi-category large margin classification methods. Journal of Machine Learning Research 5:1225–1251

Zhang T (2013) Multi-stage convex relaxation for feature selection. Bernoulli 19(5B):2277–2293

Zhang X, Aybat NS, Gürbüzbalaban M (2021) Robust accelerated primal-dual methods for computing saddle points. SIAM J Optim 34:1097–1130, URL https://api.semanticscholar.org/CorpusID:244709301

Zhang X, Aybat N, Gurbuzbalaban M (2022) SAPD+: An accelerated stochastic method for nonconvex-concave minimax problems. In: Oh AH, Agarwal A, Belgrave D, Cho K (eds) Advances in Neural Information Processing Systems, URL https://openreview.net/forum?id=GiUpEVQmNx8

Zhang Z, Lan G (2024) Optimal methods for convex nested stochastic composite optimization: Optimal methods for convex nested... Math Program 212(1):1–48, DOI 10.1007/s10107-024-02090-3, URL https://doi.org/10.1007/s10107-024-02090-3

Zhou L, Wang B, Thai MT, Yang T (2025) Stochastic primal-dual double block-coordinate for two- way partial AUC maximization. Transactions on Machine Learning Research URL https://openreview.net/forum?id=M3kibBFP4q

Zhu D, Li G, Wang B, Wu X, Yang T (2022a) When AUC meets DRO: optimizing partial AUC for deep learning with non-convex convergence guarantee. In: Chaudhuri K, Jegelka S, Song L, Szepesvári C, Niu G, Sabato S (eds) International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA, PMLR, Proceedings of Machine Learning Research, vol 162, pp 27548–27573, URL https://proceedings.mlr.press/v162/zhu22g.html

Zhu D, Li G, Wang B, Wu X, Yang T (2022b) When auc meets dro: Optimizing partial auc for deep learning with non-convex convergence guarantee. ArXiv abs/2203.00176, URL https://api.semanticscholar.org/CorpusID:247187969

Zhu D, Wu X, Yang T (2022c) Benchmarking deep AUROC optimization: Loss functions and algorithmic choices. CoRR abs/2203.14177, DOI 10.48550/ARXIV.2203.14177, URL https://doi.org/10.48550/arXiv.2203.14177, 2203.14177

Zhu D, Wang B, Chen Z, Wang Y, Sonka M, Wu X, Yang T (2023a) Provable multi-instance deep AUC maximization with stochastic pooling. In: Krause A, Brunskill E, Cho K, Engelhardt B, Sabato S, Scarlett J (eds) International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA, PMLR, Proceedings of Machine Learning Research, vol 202, pp 43205–43227, URL https://proceedings.mlr.press/v202/zhu23l.html

Zhu D, Ying Y, Yang T (2023b) Label distributionally robust losses for multi-class classification: Consistency, robustness and adaptivity. In: Krause A, Brunskill E, Cho K, Engelhardt B, Sabato S, Scarlett J (eds) International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA, PMLR, Proceedings of Machine Learning Research, vol 202, pp 43289–43325, URL https://proceedings.mlr.press/v202/zhu23o.html

Zhu L, Gürbüzbalaban M, Ruszczyński A (2023c) Distributionally robust learning with weakly convex losses: Convergence rates and finite-sample guarantees. URL https://arxiv.org/abs/2301.06619, 2301.06619

Zinkevich M (2003) Online convex programming and generalized infinitesimal gradient ascent. In: Proceedings of the Twentieth International Conference on International Conference on Machine Learning, AAAI Press, ICML'03, p 928–935

Zou H, Hastie T (2003) Regularization and variable selection via the elastic net. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 67(2):301–320