Faster Stochastic Variance Reduction Methods for Compositional MiniMax Optimization

Jin Liu[1], Xiaokang Pan[1], Junwen Duan[1], Hong-Dong Li[1], Youqi Li[2], Zhe Qu[1]*

[1]School of Computer Science and Engineering, Central South University, Changsha, China.
[2]School of Computer Science and Technology, Beijing Institute of Technology, Beijing, China.
{liujin06, 224712176, jwduan, hongdong, zhe_qu}@csu.edu.cn, liyouqi@bit.edu.cn

Abstract

This paper delves into the realm of stochastic optimization for compositional minimax optimization—a pivotal challenge across various machine learning domains, including deep AUC and reinforcement learning policy evaluation. Despite its significance, the problem of compositional minimax optimization is still under-explored. Adding to the complexity, current methods of compositional minimax optimization are plagued by sub-optimal complexities or heavy reliance on sizable batch sizes. To respond to these constraints, this paper introduces a novel method, called Nested STOchastic Recursive Momentum (NSTORM), which can achieve the optimal sample complexity and obtain the nearly accuracy solution, matching the existing minimax methods. We also demonstrate that NSTORM can achieve the same sample complexity under the Polyak-Lojasiewicz (PL)-condition—an insightful extension of its capabilities. Yet, NSTORM encounters an issue with its requirement for low learning rates, potentially constraining its real-world applicability in machine learning. To overcome this hurdle, we present ADAptive NSTORM (ADA-NSTORM) with adaptive learning rates. We demonstrate that ADA-NSTORM can achieve the same sample complexity but the experimental results show its more effectiveness. All the proposed complexities indicate that our proposed methods can match lower bounds to existing minimax optimizations, without requiring a large batch size in each iteration. Extensive experiments support the efficiency of our proposed methods.

Introduction

In recent years, minimax optimization theory has been considered more attractive due to the broad range of machine learning applications, including generative adversarial networks (Goodfellow et al. 2014; Arjovsky, Chintala, and Bottou 2017; Gulrajani et al. 2017), adversarial training of deep neural networks (Madry et al. 2018; Wang et al. 2021; Qu et al. 2023), robust optimization (Chen et al. 2017; Mohri, Sivek, and Suresh 2019; Qu et al. 2022), and policy evaluation on reinforcement learning (Sutton and Barto 2018; Hu et al. 2019; Zhang et al. 2021). At the same time, many machine learning problems can be formulated as compositional optimizations, for example, model agnostic meta-learning (Finn, Abbeel, and Levine 2017; Gao, Li, and Huang 2022) and risk-averse portfolio optimization (Zhang and Lan 2020; Shapiro, Dentcheva, and Ruszczynski 2021). Due to the important growth of these two problems in machine learning fields, the compositional minimax problem should be also clearly discussed, which can be formulated as follows:

$$\min_{x\in\mathcal{X}}\max_{y\in\mathcal{Y}} f(g(x),y) \triangleq \mathbb{E}_{\zeta} f(\mathbb{E}_{\xi}[g(x;\xi)],y;\zeta), \qquad (1)$$

where $g(\cdot) : \mathcal{X} \to \mathbb{R}^d, f(\cdot,\cdot) : (\mathbb{R}^d,\mathcal{Y}) \to \mathbb{R}$, $\xi \in \Xi$, $\zeta \in \Omega$, $\mathcal{X}$ and $\mathcal{Y}$ are convex and compact sets. Suppose that $f(g(x),y)$ is a strongly concave objective function with respect to y for all $x \in \mathcal{X}$.

Numerous research studies have been dedicated to investigating the convergence analysis of minimax optimization problems (Nemirovski et al. 2009; Palaniappan and Bach 2016; Lin, Jin, and Jordan 2020; Yang, Kiyavash, and He 2020; Chen et al. 2020; Rafique et al. 2022) across diverse scenarios. Various methodologies have been devised to address these challenges. Approaches such as Stochastic Gradient Descent Ascent (SGDA) have been proposed (Lin, Jin, and Jordan 2020), accompanied by innovations like variance-reduced SGDA (Luo et al. 2020; Xu et al. 2020) that aim to expedite convergence rates. Moreover, the application of Riemannian manifold-based optimization has been explored (Huang, Gao, and Huang 2020) across different minimax scenarios, showcasing the breadth of methodologies available. However, all of these methods are only designed for the non-compositional problem. It indicates that the stochastic gradient can be assumed as an unbiased estimation of the full gradient of both the two sub-problems. Because it is too difficult to get an unbiased estimation in compositional optimization, these methods cannot be directly used to optimize the compositional minimax problem.

Recent efforts have yielded just two studies on the non-convex compositional minimax optimization problem (1), named Stochastic Compositional Gradient Descent Ascent (SCGDA) (Gao et al. 2021) and Primal-Dual Stochastic Compositional Adaptive (PDSCA) (Yuan et al. 2022). However, they only can obtain the sample complexity $O(\kappa^4/\epsilon^4)$ for achieving the ϵ-accuracy solution, which limits the applicability in many machine learning scenarios. Consequently, there is a pressing need to devise a more streamlined approach capable of tackling this challenge. In addition, some

*Corresponding Author.

compositional minimization optimizations have been proposed, such as SCGD (Wang, Fang, and Liu 2017), STORM (Cutkosky and Orabona 2019), and RECOVER (Qi et al. 2021). They may not be directly utilized for the minimax problem (1), because the minimization of objective $f(g(x))$ depends on the maximization of objective $f(g(x), \cdot)$ for any $x \in \mathcal{X}$. Furthermore, the combined errors from Jacobian and gradient estimators worsen challenges in both sub-problems. We aim to develop an approach effectively tackling the compositional minimax problem (1), optimizing sample complexities efficiently, without requiring a large batch size.

In this paper, to address the aforementioned challenges, we first develop a novel Nested STOchastic Recursive Momentum (NSTORM) method for the problem (1). The NSTORM method leverages the variance reduction technique (Cutkosky and Orabona 2019) to estimate the inner/outer functions and their gradients. The theoretical result shows that our proposed NSTORM method can achieve the optimal sample complexity of $O(\kappa^3/\epsilon^3)$. To the best of our knowledge, NSTORM is the first method to match the best sample complexity in existing minimax optimization studies (Huang, Wu, and Hu 2023; Luo et al. 2020) without requiring a large batch size. We also demonstrate that NSTORM can achieve the same sample complexity under the Polyak-Łojasiewicz (PL)-condition, which indicates an insightful extension of NSTORM. In particular, the central idea of our proposed NSTORM method and analysis has two aspects: 1) the variance reduction is applied to both function and gradient values, which is different from (Gao et al. 2021; Yuan et al. 2022) and 2) the estimator of the inner gradient $\nabla g(x)$ is updated with a projection to ensure that the error can be bounded regardless of the minimization sub-problem. Furthermore, because NSTORM requires a small learning rate to obtain the optimal sample complexity, it may be difficult to set in real-world scenarios. To address this issue, we take advantage of adaptive learning rates in NSTORM and design an adaptive version, called ADAptive NSTORM (ADA-NSTORM). We also demonstrate that ADA-NSTORM can also obtain the same sample complexity as NSTORM, i.e., $O(\kappa^3/\epsilon^3)$ and performs better in practice without tuning the learning rate manually.

Related Work

Compositional minimization problem: The compositional minimization optimization problem is common in many real-world machine learning scenarios, e.g., meta-learning (Finn, Abbeel, and Levine 2017) and risk-averse portfolio optimization (Zhang and Lan 2020), which can be defined as follows:

$$\min_x f(g(x)) \triangleq \mathbb{E}_\zeta f(\mathbb{E}_\xi[g(x;\xi)];\zeta). \tag{2}$$

A typical challenge to optimizing the compositional minimization problem is that we cannot obtain an unbiased estimation of the full gradient by SGD, i.e., $\mathbb{E}_{\xi,\zeta}[\nabla g(x;\xi)^\top \nabla_g f(g(x;\xi),\zeta)] \neq \nabla g(x)^\top \nabla_g f(g(x))$. To address this issue, some methods have been developed in the past few years. For example, (Wang, Fang, and Liu 2017) uses stochastic gradient for the inner function value when computing the stochastic gradient. However, the convergence rate only can achieve $O(1/\epsilon^8)$ for the nonconvex objective, which has an obvious convergence gap to the regular SGD method. To improve the convergence speed, some advanced variance reduction techniques have been leveraged into Stochastic Compositional Gradient Descent (SCGD) (Wang, Fang, and Liu 2017). For example, SAGA (Zhang and Xiao 2019a), SPIDER (Fang et al. 2018), and STORM (Cutkosky and Orabona 2019) were leveraged into SCGD and achieved a better convergence result, i.e., $O(1/\epsilon^3)$. Recently, some studies (Yuan, Lian, and Liu 2019; Zhang and Xiao 2021; Jiang et al. 2022; Tarzanagh et al. 2022) bridged the gap between stochastic bilevel or multi-level optimization problems and stochastic compositional problems, and developed efficient methods. However, all of these methods only investigated the convergence result for minimization problems, ignoring the maximization sub-problem.

Minimax optimization problem: The minimax optimization problem is an important type of model and leads to many machine learning applications, e.g., adversarial training and policy optimization. Typically, the minimax optimization problem can be defined as follows:

$$\min_{x\in\mathbb{R}^{d_1}} \max_{y\in\mathcal{Y}} f(x,y) \triangleq \mathbb{E}_\xi f(x,y;\xi). \tag{3}$$

Note that both x and y in (3) are trained from the same dataset. Currently, the prevailing approach for solving minimax optimization problems involves alternating between optimizing the minimization and maximization sub-problems. Stochastic Gradient Descent Ascent (SGDA) methods (Lin, Jin, and Jordan 2020; Yan et al. 2020; Yuan and Hu 2020) have been proposed as initial solutions to address this problem. Subsequently, accelerated gradient descent ascent methods (Luo et al. 2020; Xu et al. 2020) emerged, leveraging variance reduction techniques to tackle stochastic minimax problems based on the variance reduction techniques. Additionally, research efforts have been made to explore non-smooth nonconvex-strongly-concave minimax optimization (Huang, Gao, and Huang 2020; Chen et al. 2020). Moreover, (Huang, Gao, and Huang 2020) proposed the Riemannian stochastic gradient descent ascent method and some variants for the Riemannian minimax optimization problem. (Qiu et al. 2020) reformulated nonlinear temporal-difference learning as a minimax optimization problem and proposed the single-timescale SGDA method. However, all of these methods fail to address the compositional structure inherent in the compositional minimax optimization problem presented in (1).

The Proposed Method

Design Challenge

Compared to the conventional minimax optimization problem, the main challenge in compositional minimax optimization is that we cannot obtain the unbiased gradient of the objective function f. Although we can access the unbiased estimation of each function and its gradient, i.e., $\mathbb{E}_\xi[g(x;\xi)] = g(x)$, $\mathbb{E}_\zeta[f(y;\zeta)] = f(y)$ and $\mathbb{E}_\zeta[\nabla f(y;\zeta)] = \nabla f(y)$, it is still difficult to obtain an unbiased estimation of the gradient $\nabla f(g(x), y)$. This is due to

the fact that the expectation over ξ cannot be moved into the gradient ∇f, i.e., $\mathbb{E}_{\xi,\zeta}[\nabla f(g(x;\xi),y;\zeta)] \neq \nabla f(g(x),y)$. Similarly, we cannot get the unbiased estimation of the function value f such that $\mathbb{E}_{\xi,\zeta}[f(g(x;\xi),y;\zeta)] \neq f(g(x),y)$.

Motivated by the aforementioned challenge, one potential approach to improve the evaluation of both function values and Jacobians is to utilize variance-reduced estimators. These estimators can effectively reduce estimation errors. However, applying variance-reduced estimators directly to minimax optimization in compositional minimization (Zhang and Lan 2020; Qi et al. 2021) is not straightforward. This is because if the estimators for Jacobians are not bounded, the estimation error may increase for the maximization sub-problem. In order to address this issue, (Gao et al. 2021) and (Yang, Zhang, and Fang 2022) have developed SCGDA and PDSCA methods to approach the compositional minimax optimization, respectively. However, they only obtain the sample complexity as $O(\kappa^4/\epsilon^4)$ to achieve ϵ-accuracy solution, which is much slower than existing compositional minimization or minimax optimization methods. To obtain the optimal sample complexity without requiring large batch sizes, our proposed method modifies the STORM (Cutkosky and Orabona 2019; Jiang et al. 2022) estimator and incorporates gradient projection techniques. This modification ensures that the Jacobians can be bounded for the minimization sub-problem and the gradients are projected onto a convex set for the maximization problem, thereby reducing gradient estimation errors.

Nested STOchastic Recursive Momentum (NSTORM)

In this subsection, we will present our proposed method, named the Nested STOchastic Recursive Momentum (NSTORM), to solve the compositional minimax problem in (1). We aim to find an ϵ-accuracy to achieve low sample complexity without using large batch sizes.

Our proposed NSTORM method is illustrated in Algorithm 1. Inspired by STROM (Cutkosky and Orabona 2019), the NSTORM method leverages similar variance-reduced estimators for both the two sub-problems in (1). Note that our goal is to find an ϵ-stationary point with low sample complexity. As we mentioned before because we cannot obtain the unbiased estimation of $\nabla_x f(g(x_t),y_t)$, we use estimators u_t and v'_t to estimate the inner function $g(x_t)$ and its gradient $\nabla g(x_t)$, respectively. In each iteration t, the two estimators u_t and v'_t can be computed by:

$$u_t = (1-\beta_t)u_{t-1} + \beta_t g(x_t;\xi_t) + (1-\beta_t)(g(x_t;\xi_t) - g(x_{t-1};\xi_t)), \tag{4}$$

$$v'_t = \Pi_{C_g}[(1-\beta_t)v'_{t-1} + \beta_t \nabla g(x_t;\xi_t) + (1-\beta_t)(\nabla g(x_t;\xi_t) - \nabla g(x_{t-1};\xi_t))], \tag{5}$$

where $0 < \beta_t < 1$. Note that the projection operation $\Pi_{C_g}(x) = \arg\min_{\|w\|\leq C_g} \|w - x\|^2$ aims to bound the error of the stochastic gradient estimator, which also facilitates the outer level estimator. More specifically, we need to reduce the variance of the estimator (because true gradients are in the projected domain, projection does not degrade the analysis); on the other side, we must avoid the variance of the estimator accumulating after the outer level, i.e., maximization sub-problem.

Algorithm 1: Illustration of NSTORM method.

Initialization: x_1, $y_1 = y^*(x_1)$, γ, β_t, α_t, η_t

1: **for** $t = 1$ to T **do**
2: Draw a sample ξ_t;
3: **if** $t = 1$ **then**
4: $u_t = g(x_t;\xi_t)$, $v'_t = \nabla_x g(x_t;\xi_t)$, $v''_t = \nabla_g f(u_t,y_t;\xi_t)$, and $w_t = \nabla_y f(u_t,y_t;\xi_t)$;
5: **else**
6: Compute estimators u_t and v'_t by (4) and (5);
7: Draw another sample ζ_t;
8: Compute the estimator v''_t by (6);
9: $v_t = v'_t v''_t$;
10: Compute the estimator w_t by (7);
11: **end if**
12: Update x_{t+1} and y_{t+1} by (8);
13: **end for**

For the outer level function, if we use the same strategy to compute the gradient as SCGDA (Gao et al. 2021), i.e., $v_t = (v'_t)^\top \nabla_g f(u_t,y_t;\zeta_t)$, we have to use large batches and the variance produced by v_t cannot be bounded. Therefore, we also estimate the outer function by the NSTORM method, which results in a tighter bound for $\mathbb{E}[\|v_t - \nabla_x f(g(x_t),y_t)\|^2]$. As such, we estimate the gradient $\nabla_g f(u_t,y_t)$ by v''_t, which can be computed by:

$$v''_t = (1-\beta_t)v''_{t-1} + \beta_t \nabla_g f(u_t,y_t;\zeta_t) + (1-\beta_t)(\nabla_g f(u_t,y_t;\zeta_t) - \nabla_g f(u_{t-1},y_{t-1};\zeta_t)). \tag{6}$$

Based on the chain rule, the estimated compositional gradient is equal to $v'_t v''_t$, i.e., $v_t = v'_t v''_t$. To avoid using large batches, we estimate the outer function $\nabla_y f(g(x_t),y_t)$ by w_t based on the NSTORM estimator, which can be computed by:

$$w_t = (1-\alpha_t)w_{t-1} + \alpha_t \nabla_y f(u_t,y_t;\zeta_t) + (1-\alpha_t)(\nabla_y f(u_t,y_t;\zeta_t) - \nabla_y f(u_{t-1},y_{t-1};\zeta_t)), \tag{7}$$

where $0 < \alpha_t < 1$. After obtaining the estimators v_t and w_t, we can use the following strategy to update the parameters x and y in the compositional minimax problem:

$$x_{t+1} = x_t - \gamma\eta_t v_t, \quad y_{t+1} = y_t + \eta_t w_t, \tag{8}$$

where γ is the step size, and η_t is the learning rate. Note that in the first iteration, we evaluate all estimators u_1, v'_1, v''_1, w_1 by directly computing inner level function and gradients, i.e., line 4 in Algorithm 1. The reason we choose two level estimators is to avoid using large batches, and only need to draw two samples, i.e., ξ_t and ζ_t, to calculate estimators for updating x_t and y_t, respectively. The common idea to achieve the optimal solution of minimax (Lin, Jin, and Jordan 2020) is that the step size of x should be smaller than y. In addition, the compositional minimization sub-problem will generate larger errors, which incurs more challenges for NSTORM. Particularly, in (8), if we simply set the same

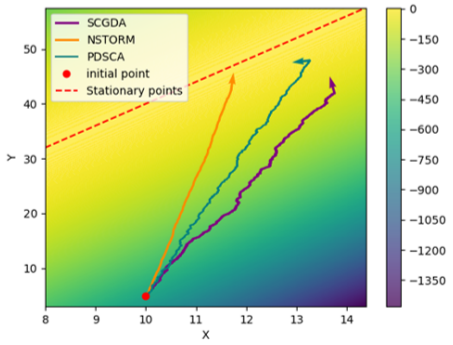

Figure 1: Trajectories of different methods for the compositional minimax optimization.

step sizes of x and y, our proposed NSTORM method may fail to converge, which is confirmed in the subsequent proof. Therefore, we set $\gamma < 1$ to ensure that the step size of x is less than y.

To clearly explain the advantages of NSTORM, a toy example is illustrated in Figure 1. Consider the following concrete example of a nonconvex-strongly-concave function.: $f(g(x), y) = -2g(x)^2 + 2g(x)y - \frac{1}{2}y^2$, where $g(x) = 2x$. In Figure 1, we simulate these stochastic oracles by adding noise when obtaining function gradients and function values. This function obtains the biased estimation in the minimization sub-problem affording the problem in (1). It can be observed that NSTORM performs more robustness on noisy and biased estimation, which brings up an opportunity to obtain the optimal solution with shorter and smoother paths compared to other benchmarks.

Convergence Analysis of NSTORM

In what follows, we will prove the convergence rate of our proposed NSTROM method in Algorithm 1. We first state some commonly-used assumptions for compositional and minimax optimizations (Gao et al. 2021; Wang, Fang, and Liu 2017; Xian et al. 2021; Yuan, Lian, and Liu 2019; Zhang and Lan 2020) to facilitate our convergence analysis. In order to simplify the notations and make the paper coherence, we denote $\nabla f(a,b) = (\nabla_a f(a,b), \nabla_b f(a,b))$ for $(a,b) \in \mathcal{A} \times \mathcal{B}$ in the following assumptions, where $\mathcal{A} = \{g(x) | x \in \mathcal{X}\}$ and $\mathcal{B} = \mathcal{Y}$.

Assumption 1. *(Smoothness) There exists a constant $L > 0$, such that*

$$\|\nabla f(a_1,b_1) - \nabla f(a_2,b_2)\| \le L\|(a_1,b_1) - (a_2,b_2)\|,$$

where $\forall (a_1,b_1),(a_2,b_2) \in \mathcal{A} \times \mathcal{B}$. In addition, we assume that there exists a constant $L_g > 0$, $L_f > 0$ such that

$$\|\nabla g(x_1;\xi) - \nabla g(x_2;\xi)\| \le L_g\|x_1 - x_2\|$$
$$\|g(x_1;\xi) - g(x_2;\xi)\| \le L_f\|x_1 - x_2\|,$$

where $\forall x_1, x_2 \in \mathcal{X}$.

Assumption 2. *(Bounded Gradient) There exist two constants $C_g > 1$ and $C_f > 0$, where the two gradients can be bounded by $\mathbb{E}[\|\nabla g(x)\|^2] \le C_g^2$, $\forall x \in \mathcal{X}$ and $\mathbb{E}[\|\nabla f(a,b)\|^2] \le C_f^2$, $\forall (a,b) \in \mathcal{A} \times \mathcal{B}$.*

Assumption 3. *(Bounded Variance) There exist three constants $\sigma_f > 0$, $\sigma_g > 0$, and $\sigma_{g'} > 0$, where the three kinds of variance can be bounded by:*

$$\mathbb{E}[\|\nabla f(a,b;\zeta) - \nabla f(a,b)\|^2] \le \sigma_f^2, \ \forall (a,b) \in \mathcal{A} \times \mathcal{B},$$
$$\mathbb{E}[\|\nabla g(x;\xi) - \nabla g(x)\|^2] \le \sigma_{g'}^2, \ x \in \mathcal{X},$$
$$\mathbb{E}[\|g(x;\xi) - g(x)\|^2] \le \sigma_g^2, \ x \in \mathcal{X}.$$

Assumption 4. *(Strongly Concave) There exists a constant $\mu > 0$, such that*

$$f(a,b_1) \le f(a,b_2) + \langle \nabla_b f(a,b_2), a_1 - b_2\rangle - \frac{\mu}{2}\|b_1 - b_2\|^2,$$

where $\forall a \in \mathcal{A}$ and $\forall b_1, b_2 \in \mathcal{B}$.

Similar to existing minimax studies (Lin, Jin, and Jordan 2020; Xian et al. 2021), we also use ϵ-point of $\nabla\Phi(x)$, i.e., $\|\nabla\Phi(x)\| \le \epsilon$ as the convergence criterion in our focused compositional minimax problem, where $\Phi(x) = \max_{y\in\mathcal{Y}} f(g(x),y)$ and $y^*(x) = \arg\max_{y\in\mathcal{Y}} f(g(x),y)$. We demonstrate that $\Phi(x)$ is differentiable and $(C_g^2 L\kappa + C_f L_g)$-smooth, where $\kappa = L/\mu$, and $y^*(x)$ is κ-Lipschitz, which has some differences compared to the minimax optimization (Lin, Jin, and Jordan 2020).

Now, we can obtain the following convergence result of our proposed NSTORM method in Algorithm 1 to solve the compositional minimax problem in (1):

Theorem 1. *Under the Assumptions 1-4, for Algorithm 1, by setting $\eta_t = \frac{1}{(m+t)^{1/3}}$, $m > \max\{125L^3, 8\gamma^3 L_\Phi^3, (12L^2c_1^2 + 4L^2c_2^2)^3, c_1^3, c_2^3\}$, $c_1 \ge 2 + 4\gamma(C_f^2 + C_g^2) + 2C_g^2L^2\gamma$, $c_2 \ge \frac{2}{3} + 180L^2 + \frac{36\gamma C_g^2 L^2}{\mu^2}$, $\beta_t = c_1\eta_{t-1}^2$, $\alpha_t = c_2\eta_{t-1}^2$, $0 < \gamma \le \frac{1}{\sqrt{B^2 + 20\kappa^4 C_g^2}}$, where $B = \frac{100C_g^2L^4}{\mu^2} + 2L_f^2 + 2L_g^2 + 12L^2L_f^2 + 4L^2C_g^2$, we can obtain the following:*

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\|\nabla\Phi(x_t)\| \le \frac{m^{1/6}\sqrt{M}}{\sqrt{\gamma T}} + \frac{\sqrt{M}}{\sqrt{\gamma}T^{1/3}},$$

where $M = \Phi(x_1) - \Phi_* + \sigma_g^2 + \sigma_{g'}^2 + \sigma_f^2 + L^2\sigma_g^2 + (c_1^2(2\sigma_g^2 + 2\sigma_{g'}^2 + 2\sigma_f^2 + 4L^2\sigma_g^2) + 4\sigma_f^2c_2^2)\ln(T+m)$ and Φ_* represents the minimum value of $\Phi(x)$.

Remark 1. As discussed in the previous section, our proposed NSTORM method in Algorithm 1 results in tighter bounds for all variances, i.e., $\mathbb{E}[\|u_t - g(x_t)\|^2]$, $\mathbb{E}[\|v_t' - \nabla g(x_t)\|^2]$, $\mathbb{E}[\|v_t'' - \nabla_g f(u_t,y_t)\|^2]$ and $\mathbb{E}[\|w_t - \nabla_y f(g(x_t),y_t)\|^2]$, which makes NSTROM method converge faster comparing with existing studies. Therefore, it is very important to show the upper bounds of these variances.

Remark 2. Without loss of generality, let $m = O(1)$, we have $M = O(\ln(m+T)) = O(1)$. Therefore, our proposed NSTORM method has a convergence rate of $O\left(1/T^{1/3}\right)$. Let $\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[\nabla\Phi(x_t)] = O\left(1/T^{1/3}\right) \le \epsilon$, we have $T = O\left(\kappa^3/\epsilon^3\right)$. Because we only need two samples, i.e., $O(1)$, to estimate the stochastic to compute the gradient in each iteration, and need T iterations. Therefore, our NSTORM

Algorithm 2: Illustration of ADA-NSTORM method.

Initialization: x_1, $y_1 = y^*(x_1)$, γ, λ, β_t, α_t, η_t, τ, a_t, b_t.

1: **for** $t = 1$ to T **do**
2: Draw a sample ξ_t;
3: **if** $t = 1$ **then**
4: $u_t = g(x_t;\xi_t)$, $v_t' = \nabla_x g(x_t;\xi_t)$, $v_t'' = \nabla_g f(u_t, y_t;\xi_t)$, and $w_t = \nabla_y f(u_t, y_t;\xi_t)$;
5: **else**
6: Compute the estimator u_t by (4);
7: Compute the estimator v_t' by (5);
8: Draw another sample ζ_t;
9: Compute the estimator v_t'' by (6);
10: $v_t = v_t' v_t''$;
11: Compute the estimator w_t by (7);
12: **end if**
13: Generate the adaptive matrices $A_t \in \mathbb{R}^{d\times d}$ and $B_t \in \mathbb{R}^{p\times p}$ by (9) and (10) (Adam);
14: Compute the $\tilde{x}_{t+1}$ and $\tilde{y}_{t+1}$ by (11) and (12);
15: Compute the x_{t+1} and y_{t+1} by (13);
16: **end for**

method requires sample complexity of $O\left(\kappa^3/\epsilon^3\right)$ for finding an ϵ-accuracy point of the compositional minimax problem in (1). Because the SCGDA method (Gao et al. 2021) only achieves $O\left(\kappa^4/\epsilon^4\right)$ with requiring a large batch size as $O(T)$, it is observed that our proposed NSTROM method improves the convergence rate significantly.

Remark 3. It is worth noting that if we moderate the assumption of $f(g(x), y)$ with respect to y to follow the PL-condition instead of strongly-concave in Assumption 4, NSTORM can also obtain the sample complexity, i.e., $O(\kappa^3/\epsilon^3)$. To the best of our knowledge, this is the first study to design the method for compositional minimax optimization, which highlights the extensibility and applicability of NSTORM.

ADAptive-NSTORM (ADA-NSTORM)

Learning Procedure of ADA-NSTORM

According to the analysis of variance in the NSTORM method, due to the large variance of the two-level estimator, we must select a smaller learning step to update parameters x and y. As a result, this degrades the applicability of NSTORM. Adaptive learning rates (Huang, Gao, and Huang 2020; Huang, Wu, and Hu 2023) have been developed to accelerate many optimization methods including (stochastic) gradient-based methods based on momentum technology. Therefore, we leverage adaptive learning rates in NSTORM and propose ADAptive NSTORM (ADA-NSTORM) method, which is illustrated in Algorithm 2.

In each iteration t, we first use the NSTORM method to update all estimators related to the inner/outer functions and their gradients. At Line 13 in Algorithm 2, we generate the adaptive matrices A_t and B_t for the two variables x and y, respectively. In particular, the general adaptive matrix $A_t \succeq \rho I_d$ is updated for the variable x, and the global adaptive matrix B_t is for y. It is worth noting that we can generate the two matrices A_t and B_t by a class of adaptive learning rates generators such as Adam (Kingma and Ba 2014), AdaBelief, (Zhuang et al. 2020), AMSGrad (Reddi, Kale, and Kumar 2018), AdaBound (Luo, Xiong, and Liu 2019). In particular, the Adam generator can be computed by:

$$a_t = \tau a_{t-1} + (1-\tau)v_t^2, \;\; A_t = \mathrm{diag}\left(\sqrt{a_t} + \rho\right), \quad (9)$$

$$b_t = \tau b_{t-1} + (1-\tau)w_t^2, \;\; B_t = \mathrm{diag}(\sqrt{b_t} + \rho), \quad (10)$$

where $t \geq 1$, $\tau \in (0,1)$ and $\rho > 0$. Due to the biased full gradient in the compositional minimax problem, we leverage the gradient estimator v_t and w_t to update adaptive matrices instead of simply using the gradient, i.e., $\nabla_x f(x_t, y_t;\xi_t)$ and $\nabla_y f(x_t, y_t;\zeta_t)$ (Huang, Gao, and Huang 2020; Huang, Wu, and Hu 2023). After obtaining adaptive learning matrices A_t and B_t, we use adaptive stochastic gradient descent to update the parameters x and y as follows:

$$\begin{aligned}\tilde{x}_{t+1} &= x_t - \gamma A_t^{-1} v_t \\ &= \arg\min_{x\in\mathbb{R}^d} \left\{ \langle x, v_t\rangle + \frac{1}{2\gamma}\left(x - x_t\right)^T A_t \left(x - x_t\right) \right\}, \quad (11)\end{aligned}$$

$$\begin{aligned}\tilde{y}_{t+1} &= y_t - \lambda B_t^{-1} w_t \\ &= \arg\max_{y\in\mathbb{P}^d} \left\{ \langle y, w_t\rangle + \frac{1}{2\lambda}\left(y - y_t\right)^T B_t \left(y - y_t\right) \right\}, \quad (12)\end{aligned}$$

where γ and λ are step sizes for updating $\tilde{x}$ and $\tilde{y}$, respectively. At Line 15 in Algorithm 2, we use the momentum iteration to further update the primal variable x and the dual variable y as follows:

$$\begin{aligned} x_{t+1} &= x_t + \eta_t(\tilde{x}_{t+1} - x_t), \\ y_{t+1} &= y_t + \eta_t(\tilde{y}_{t+1} - y_t). \end{aligned} \quad (13)$$

Convergence Analysis of ADA-STORM

We will introduce one additional assumption to facilitate the convergence analysis of ADA-NSTORM.

Assumption 5. *In Algorithm 2, the adaptive matrices A_t, $\forall t \geq 1$ for updating the variables x satisfies $A_t^T = A_t$ and $\lambda_{min}(A_t) \geq \rho > 0$, where ρ is an appropriate positive number. We consider the adaptive matrics B_t, $\forall t \geq 1$ for updating the variables y satisfies $\hat{b} I_p \geq B_t \geq b I_p > 0$, where I_p denotes a d-dimensional identity matrix.*

Remark 4. Assumption 5 ensures that the adaptive matrices A_t, $\forall t \geq 1$ are positive definite, which is widely used in (Huang, Gao, and Huang 2020; Huang, Wu, and Hu 2023; Huang 2023). This Assumption also guarantees that the global adaptive matrices B_t, $\forall t \geq 1$ are positive definite and bounded, resulting in mild conditions. To support the mildness of this assumption, we will empirically show that the learning performance does not have obvious changes by varying the bound of a_t and b_t.

Theorem 2. *Given Assumptions 1-5, for Algorithm 2, by setting* $\eta_t = \frac{1}{(m+t)^{1/3}}$, $m > \max\{\frac{8L_\Phi^3\gamma^3}{\rho^3}, (10L^2c_1^2 + 4L^2c_2^2)^3, c_1^3, c_2^3\}$, $c_1 \geq 2 + \frac{5\gamma(2C_f^2+2C_g^2+C_g^2L^2)}{\rho}$, $c_2 \geq \frac{2}{3} + \frac{125\lambda L^2}{2\mu b} + \frac{125\gamma C_g^2\kappa^2\hat{b}}{3b}$, $\gamma \leq \frac{\rho}{4\sqrt{B_1^2+\rho B_2}}$, *where* $B_1 = \frac{50C_g^2\kappa^4\hat{b}}{\lambda^2}$, $B_2 = \frac{70\kappa^3 L}{\lambda} + 2L_f^2 + 2L_g^2 + 12L^2L_f^2 + 4L^2C_g^2$, $\beta_{t+1} =$

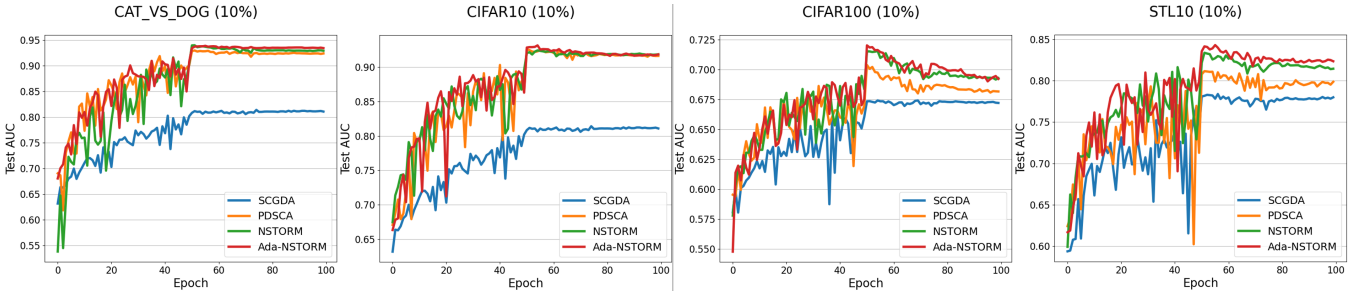

Figure 2: Convergence performance on four benchmark datasets with an imbalance ratio of 10%

$c_1\eta_t^2 \leq c_1\eta_t < 1$, $\alpha_{t+1} = c_2\eta_t^2 \leq c_2\eta_t < 1$, $0 < \lambda \leq \frac{b}{6L}$, *we can obtain the following:*

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[\|\nabla\Phi(x_t)\|] \leq \sqrt{\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[\|A_t\|^2]}\cdot\left(\frac{2\sqrt{5M}m^{1/3}}{\sqrt{\gamma T}} + \frac{2\sqrt{5M}}{\sqrt{\gamma}T^{1/3}}\right),$$

where $M = (\Phi(x_1) - \Phi_* + \sigma_g^2 + \sigma_{g'}^2 + \sigma_f^2 + L^2\sigma_g^2)/\rho + ((2c_1^2(\sigma_g^2 + \sigma_{g'}^2 + \sigma_f^2 + 6L^2\sigma_g^2) + 4c_2^2\sigma_f^2)\ln(m+T))/\rho$ *and* Φ_* *represents the minimum value of* $\Phi(x)$.

Remark 5. Without loss of generality, let $b = O(1)$ and $\hat{b} = O(1)$. From Theorem 2, given $\gamma \leq \frac{\rho}{4\sqrt{B_1^2+\rho B_2}}$, where $B_1 = \frac{50C_g^2\kappa^4\hat{b}}{\lambda^2}$, $B_2 = \frac{70\kappa^3 L}{\lambda} + 2L_f^2 + 2L_g^2 + 12L^2L_f^2 + 4L^2C_g^2$, $\lambda \leq \frac{b}{6L}$, we can see that $\gamma = O(1/\kappa^2)$, $\lambda = O(1/L)$, $M = O(1)$. Then, we can get the convergence rate $O\left(1/T^{1/3}\right)$. Therefore, to achieve ϵ-accuracy solution, the total sample complexity is $O(\kappa^3/\epsilon^3)$. It is worth noting that the term $\sqrt{\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[\|A_t\|^2]}$ is bounded to the existing adaptive learning rates in Adam algorithm (Kingma and Ba 2014) and so on. (Yuan et al. 2022) develops a PDSCA method with adaptive learning rates to approach the compositional minimax optimization problem and achieves $O(1/\epsilon^4)$ complexity. However, (Yuan et al. 2022) sets $\eta = O(1/\sqrt{T})$, which is difficult to know in practice.

Experiments

In this section, we present the results of our experiments that assess the performance of two proposed methods: NSTORM and ADA-NSTORM in the deep AUC problem (Yuan et al. 2020, 2022). To establish a benchmark, we compare our proposed methods against existing compositional minimax methods, namely SCGDA (Gao et al. 2021) and PDSCA (Yuan et al. 2022). To optimize the AUC loss, the outer function corresponds to an AUC loss and the inner function represents a gradient descent step for minimizing a cross-entropy loss, as (Yuan et al. 2022). The deep AUC problem can be formulated as follows:

$$\min_{x,a,b}\max_{y\in\Omega}\Theta\left(x - \alpha\nabla L_{\text{AVG}}(x), a, b, y\right). \tag{14}$$

The function Θ is to optimize the AUC score. The inner function $x - \alpha\nabla L_{\text{AVG}}(x)$ aims to optimize the average cross-entropy loss L_{AVG}. α is a hyper-parameter.

Rather than the deep AUC problem, we also evaluate our proposed methods on the risk-averse portfolio optimization problem (Shapiro, Dentcheva, and Ruszczynski 2021; Zhang et al. 2021) and the policy evaluation in reinforcement learning (Yuan, Lian, and Liu 2019; Zhang and Xiao 2019b).

Learning Model and Datasets. We employ four distinct image classification datasets in our study: CAT_vs_DOG, CIFAR10, CIFAR100 (Krizhevsky 2009), and STL10 (Coates, Ng, and Lee 2011). To create imbalanced binary variants prioritizing AUC optimization, we followed (Yuan et al. 2020) methodology. Similarly, as in (Yuan et al. 2022), ResNet20 (He et al. 2016) was used. Weight decay was consistently set to 1e-4. Each method was trained with batch size 128, spanning 100 epochs. We varied parameter m (50, 500, 5000) and set γ (1, 0.9, 0.5). Learning rate η_t reduced by 10 at 50% and 75% training. Also, β is set to 0.9. For robustness, each experiment was conducted thrice with distinct seeds, computing mean and standard deviations. Notably, the ablation study focused on the CIFAR100 dataset, 10% imbalanced ratio, as detailed in the main paper.

Performance Evaluation

The training progression of deep AUC is illustrated in Figure 2. It shows the notable swiftness of convergence exhibited by our two proposed methods. Furthermore, across all four datasets, our methods consistently yield the most favorable test AUC outcomes. It is evident from the results depicted in Figure 2 that even NSTORM, which lacks an adaptive generator, surpasses the performance of SCGDA and PDSCA methods, thus reinforcing the validity of our theoretical analysis. Intriguingly, despite ADA-NSTORM sharing a theoretical foundation with NSTORM, it outperforms the testing AUC performance in the majority of scenarios.

The testing AUC outcomes are summarized in Table 1, with the optimal AUC values among 100 epochs. Combining these findings with Figure 2, a recurring pattern emerges: best testing AUC performance is typically attained around the 50th epoch, followed by overfitting. Both Table 1 and Figure 2 show that our proposed methods consistently outperform benchmarks. ADA-NSTORM achieves an impressive AUC of 0.833 on STL10 with a 10% imbalanced ratio.

Datasets	CAT_vs_DOG				CIFAR10			
imratio	SCGDA	PDSCA	NSTORM	ADA-NSTORM	SCGDA	PDSCA	NSTORM	ADA-NSTORM
1%	0.750 ±0.004	**0.792 ±0.009**	0.786 ±0.001	0.786 ±0.011	0.679 ±0.011	0.699 ±0.008	0.689 ±0.003	**0.703 ±0.008**
5%	0.826 ±0.006	0.890 ±0.006	0.895 ±0.006	**0.901 ±0.004**	0.782 ±0.006	0.878 ±0.003	0.882 ±0.002	**0.894 ±0.005**
10%	0.857 ±0.009	0.932 ±0.002	0.932 ±0.005	**0.933 ±0.002**	0.818 ±0.004	0.926 ±0.001	0.926 ±0.001	**0.931 ±0.001**
30%	0.897 ±0.008	0.969 ±0.002	0.970 ±0.001	**0.972 ±0.001**	0.882 ±0.005	0.953 ±0.001	0.953 ±0.002	**0.955 ±0.001**
Datasets	CIFAR100				STL10			
imratio	SCGDA	PDSCA	NSTORM	ADA-NSTORM	SCGDA	PDSCA	NSTORM	ADA-NSTORM
1%	0.588 ±0.007	0.583 ±0.004	0.583 ±0.007	**0.593 ±0.002**	0.670 ±0.006	**0.682 ±0.016**	0.659 ±0.013	0.657 ±0.003
5%	0.641 ±0.007	0.651 ±0.006	0.648 ±0.003	**0.655 ±0.007**	0.734 ±0.007	0.775 ±0.003	0.779 ±0.005	**0.781 ±0.007**
10%	0.673 ±0.005	0.708 ±0.006	0.709 ±0.002	**0.715 ±0.006**	0.779 ±0.014	0.824 ±0.011	0.827 ±0.007	**0.833 ± 0.004**
30%	0.713 ±0.002	0.787 ±0.007	0.786 ±0.001	**0.787 ±0.001**	0.843 ±0.010	**0.901 ±0.002**	0.893 ±0.005	0.895 ±0.007

Table 1: Testing performance on the four datasets by varying imbalanced ratios.

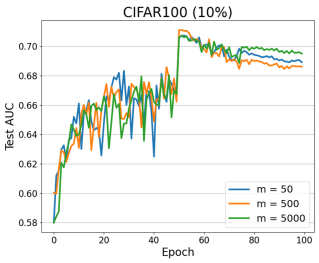

Figure 3: Impact of m for NSTORM.

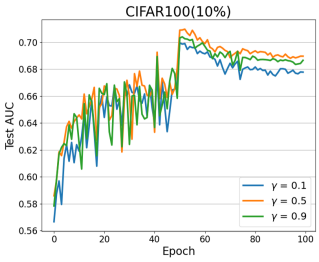

Figure 4: Impact of γ for NSTORM.

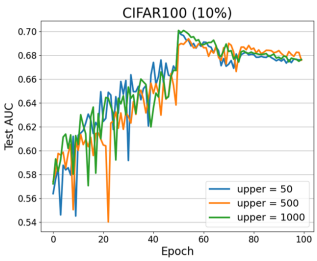

Figure 5: Impact of upper bound on a_t and b_t.

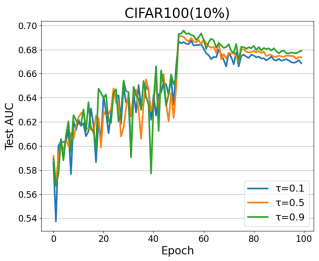

Figure 6: Impact of τ for ADA-NSTORM.

Notable exceptions are the CAT_vs_DOG and CIFAR100 datasets with a 1% imbalanced ratio, possibly due to their proximity to the training set's distribution.

Ablation Study

We conducted experiments to fine-tune parameters m and γ for NSTORM, as demonstrated in Figure 3 and Figure 4. In Theorem 1, we consider m as the lower bound, controlling the learning rate η_t. Interestingly, adjusting m yields minimal alterations. On the other hand, γ determines the relative step sizes of x and y. Figure 4 reveals that an optimal γ value of approximately 0.5 yields a testing AUC of 0.827.

To assess the influence of a_t and b_t in Assumption 5, we investigate the testing AUC under varying upper bounds for a_t and b_t, as illustrated in Figure 5. Notably, changing from an upper bound of 50 to 1000 yields a minimal change in the testing AUC, validating the mildness of Assumption 5. In addition, the parameter τ is related to the adaptive generator within ADA-NSTORM. Figure 6 shows the impact of τ on ADA-NSTORM's performance within the deep AUC problem. Intriguingly, varying τ from 0.1 to 0.9 leads to a mere change of 0.127. These ablation studies effectively reinforce the robustness of our proposed methods.

Conclusion

In this paper, we first proposed a novel method named NSTORM for optimizing the compositional minimax problem. By leveraging variance-reduced techniques of both function and gradient values, we demonstrate that the proposed NSTORM method can achieve the sample complexity of $O(\kappa^3/\epsilon^3)$ for finding an ϵ-stationary point without using large batch sizes. NSTORM under the PL-condition is also demonstrated to achieve the same sample complexity, which indicates its extendability. To the best of our knowledge, all theoretical results match the best sample complexity in existing minimax optimization. Because NSTORM requires a small learning rate to achieve the optimal complexity, this limits its applicability in real-world machine learning scenarios. To take advantage of adaptive learning rates, we develop an adaptive version of NSTORM named ADA-STORM, which can achieve the same complexity with the learning rate changing adaptively. Extensive experimental results support the effectiveness of our proposed methods.

Acknowledgments

This work of Jin Liu and Xiaokang Pan was supported in part by the National Natural Science Foundation of

China (NSFC) under Grant 62172444, in part by the Natural Science Foundation of Hunan Province under Grant 2022JJ30753, in part by the Central South University Innovation-Driven Research Programme under Grant 2023CXQD018. This work of Hongdong Li was supported in part by NSFC under Grant U22A2041. This work of Youqi Li was partially supported by NSFC under Grant 62102028, and Beijing Institute of Technology Research Fund Program for Young Scholars. This work of Zhe Qu was partially supported by NSFC under Grant 62302525 and in part by the High Performance Computing Center of Central South University.

References

Arjovsky, M.; Chintala, S.; and Bottou, L. 2017. Wasserstein generative adversarial networks. In *International conference on machine learning*, 214–223. PMLR.

Chen, C.; Luo, L.; Zhang, W.; and Yu, Y. 2020. Efficient projection-free algorithms for saddle point problems. *Advances in Neural Information Processing Systems*, 33: 10799–10808.

Chen, R. S.; Lucier, B.; Singer, Y.; and Syrgkanis, V. 2017. Robust optimization for non-convex objectives. *Advances in Neural Information Processing Systems*, 30.

Coates, A.; Ng, A.; and Lee, H. 2011. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, 215–223. JMLR Workshop and Conference Proceedings.

Cutkosky, A.; and Orabona, F. 2019. Momentum-based variance reduction in non-convex sgd. *Advances in neural information processing systems*, 32.

Fang, C.; Li, C. J.; Lin, Z.; and Zhang, T. 2018. Spider: Near-optimal non-convex optimization via stochastic path-integrated differential estimator. *Advances in Neural Information Processing Systems*, 31.

Finn, C.; Abbeel, P.; and Levine, S. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, 1126–1135. PMLR.

Gao, H.; Li, J.; and Huang, H. 2022. On the convergence of local stochastic compositional gradient descent with momentum. In *International Conference on Machine Learning*, 7017–7035. PMLR.

Gao, H.; Wang, X.; Luo, L.; and Shi, X. 2021. On the Convergence of Stochastic Compositional Gradient Descent Ascent Method. In *Thirtieth International Joint Conference on Artificial Intelligence (IJCAI)*.

Goodfellow, I. J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A. C.; and Bengio, Y. 2014. Generative Adversarial Nets. In *NIPS*.

Gulrajani, I.; Ahmed, F.; Arjovsky, M.; Dumoulin, V.; and Courville, A. C. 2017. Improved training of wasserstein gans. *Advances in neural information processing systems*, 30.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hu, W.; Li, C. J.; Lian, X.; Liu, J.; and Yuan, H. 2019. Efficient smooth non-convex stochastic compositional optimization via stochastic recursive gradient descent. *Advances in Neural Information Processing Systems*, 32.

Huang, F. 2023. Enhanced Adaptive Gradient Algorithms for Nonconvex-PL Minimax Optimization. *arXiv preprint arXiv:2303.03984*.

Huang, F.; Gao, S.; and Huang, H. 2020. Gradient descent ascent for min-max problems on riemannian manifolds. *arXiv preprint arXiv:2010.06097*.

Huang, F.; Wu, X.; and Hu, Z. 2023. Adagda: Faster adaptive gradient descent ascent methods for minimax optimization. In *International Conference on Artificial Intelligence and Statistics*, 2365–2389. PMLR.

Jiang, W.; Wang, B.; Wang, Y.; Zhang, L.; and Yang, T. 2022. Optimal algorithms for stochastic multi-level compositional optimization. *arXiv preprint arXiv:2202.07530*.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Krizhevsky, A. 2009. Learning Multiple Layers of Features from Tiny Images. *Technical report, University of Toronto*.

Lin, T.; Jin, C.; and Jordan, M. 2020. On gradient descent ascent for nonconvex-concave minimax problems. In *International Conference on Machine Learning*, 6083–6093. PMLR.

Luo, L.; Xiong, Y.; and Liu, Y. 2019. Adaptive Gradient Methods with Dynamic Bound of Learning Rate. In *International Conference on Learning Representations*.

Luo, L.; Ye, H.; Huang, Z.; and Zhang, T. 2020. Stochastic recursive gradient descent ascent for stochastic nonconvex-strongly-concave minimax problems. *Advances in Neural Information Processing Systems*, 33: 20566–20577.

Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; and Vladu, A. 2018. Towards Deep Learning Models Resistant to Adversarial Attacks. In *International Conference on Learning Representations*.

Mohri, M.; Sivek, G.; and Suresh, A. T. 2019. Agnostic federated learning. In *International Conference on Machine Learning*, 4615–4625. PMLR.

Nemirovski, A.; Juditsky, A.; Lan, G.; and Shapiro, A. 2009. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on optimization*, 19(4): 1574–1609.

Palaniappan, B.; and Bach, F. 2016. Stochastic variance reduction methods for saddle-point problems. *Advances in Neural Information Processing Systems*, 29.

Qi, Q.; Guo, Z.; Xu, Y.; Jin, R.; and Yang, T. 2021. An online method for a class of distributionally robust optimization with non-convex objectives. *Advances in Neural Information Processing Systems*, 34: 10067–10080.

Qiu, S.; Yang, Z.; Wei, X.; Ye, J.; and Wang, Z. 2020. Single-timescale stochastic nonconvex-concave optimization for smooth nonlinear TD learning. *arXiv preprint arXiv:2008.10103*.

Qu, Z.; Li, X.; Duan, R.; Liu, Y.; Tang, B.; and Lu, Z. 2022. Generalized Federated Learning via Sharpness Aware Minimization. *arXiv preprint arXiv:2206.02618*.

Qu, Z.; Li, X.; Han, X.; Duan, R.; Shen, C.; and Chen, L. 2023. How To Prevent the Poor Performance Clients for Personalized Federated Learning? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12167–12176.

Rafique, H.; Liu, M.; Lin, Q.; and Yang, T. 2022. Weakly-convex–concave min–max optimization: provable algorithms and applications in machine learning. *Optimization Methods and Software*, 37(3): 1087–1121.

Reddi, S. J.; Kale, S.; and Kumar, S. 2018. On the Convergence of Adam and Beyond. In *International Conference on Learning Representations*.

Shapiro, A.; Dentcheva, D.; and Ruszczynski, A. 2021. *Lectures on stochastic programming: modeling and theory*. SIAM.

Sutton, R. S.; and Barto, A. G. 2018. *Reinforcement learning: An introduction*. MIT press.

Tarzanagh, D. A.; Li, M.; Thrampoulidis, C.; and Oymak, S. 2022. FedNest: Federated Bilevel, Minimax, and Compositional Optimization. In *International Conference on Machine Learning*, 21146–21179. PMLR.

Wang, J.; Zhang, T.; Liu, S.; Chen, P.-Y.; Xu, J.; Fardad, M.; and Li, B. 2021. Adversarial attack generation empowered by min-max optimization. *Advances in Neural Information Processing Systems*, 34: 16020–16033.

Wang, M.; Fang, E. X.; and Liu, H. 2017. Stochastic compositional gradient descent: algorithms for minimizing compositions of expected-value functions. *Mathematical Programming*, 161(1): 419–449.

Xian, W.; Huang, F.; Zhang, Y.; and Huang, H. 2021. A faster decentralized algorithm for nonconvex minimax problems. *Advances in Neural Information Processing Systems*, 34: 25865–25877.

Xu, T.; Wang, Z.; Liang, Y.; and Poor, H. V. 2020. Enhanced first and zeroth order variance reduced algorithms for min-max optimization.

Yan, Y.; Xu, Y.; Lin, Q.; Liu, W.; and Yang, T. 2020. Optimal epoch stochastic gradient descent ascent methods for min-max optimization. *Advances in Neural Information Processing Systems*, 33: 5789–5800.

Yang, J.; Kiyavash, N.; and He, N. 2020. Global convergence and variance reduction for a class of nonconvex-nonconcave minimax problems. *Advances in Neural Information Processing Systems*, 33: 1153–1165.

Yang, S.; Zhang, Z.; and Fang, E. X. 2022. Stochastic Compositional Optimization with Compositional Constraints. *arXiv preprint arXiv:2209.04086*.

Yuan, H.; and Hu, W. 2020. Stochastic recursive momentum method for non-convex compositional optimization. *arXiv preprint arXiv:2006.01688*.

Yuan, H.; Lian, X.; and Liu, J. 2019. Stochastic recursive variance reduction for efficient smooth non-convex compositional optimization. *arXiv preprint arXiv:1912.13515*.

Yuan, Z.; Guo, Z.; Chawla, N.; and Yang, T. 2022. Compositional Training for End-to-End Deep AUC Maximization. In *International Conference on Learning Representations*.

Yuan, Z.; Yan, Y.; Sonka, M.; and Yang, T. 2020. Robust deep auc maximization: A new surrogate loss and empirical studies on medical image classification. *arXiv preprint arXiv:2012.03173*, 8.

Zhang, J.; and Xiao, L. 2019a. A composite randomized incremental gradient method. In *International Conference on Machine Learning*, 7454–7462. PMLR.

Zhang, J.; and Xiao, L. 2019b. A stochastic composite gradient method with incremental variance reduction. *Advances in Neural Information Processing Systems*, 32.

Zhang, J.; and Xiao, L. 2021. Multilevel composite stochastic optimization via nested variance reduction. *SIAM Journal on Optimization*, 31(2): 1131–1157.

Zhang, X.; Liu, Z.; Liu, J.; Zhu, Z.; and Lu, S. 2021. Taming Communication and Sample Complexities in Decentralized Policy Evaluation for Cooperative Multi-Agent Reinforcement Learning. *Advances in Neural Information Processing Systems*, 34: 18825–18838.

Zhang, Z.; and Lan, G. 2020. Optimal algorithms for convex nested stochastic composite optimization. *arXiv preprint arXiv:2011.10076*.

Zhuang, J.; Tang, T.; Ding, Y.; Tatikonda, S. C.; Dvornek, N.; Papademetris, X.; and Duncan, J. 2020. Adabelief optimizer: Adapting stepsizes by the belief in observed gradients. *Advances in neural information processing systems*, 33: 18795–18806.