Stability and Generalization for Stochastic (Compositional) Optimizations

Xiaokang Pan[1,2], **Jin Liu**[1,2], **Hulin Kuang**[1,2,*], **Youqi Li**[3], **Lixing Chen**[4,5], **Zhe Qu**[1,2,*]
[1]School of Computer Science and Engineering, Central South University
[2]Xiangjiang Laboratory
[3]School of Computer Science and Technology, Beijing Institute of Technology
[4]School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University
[5]Shanghai Key Laboratory of Integrated Administration Technologies for Information Security
{224712176, liujin06, hulinkuang, zhe_qu}@csu.edu.cn, liyouqi@bit.edu.cn, lxchen@sjtu.edu.cn

Abstract

The use of estimators instead of stochastic gradients for updates has been shown to improve algorithm convergence rates of, but their impact on generalization remains under-explored. In this paper, we investigate how estimators influence generalization. Our focus is on two widely studied problems: stochastic optimization (SO) and stochastic compositional optimization (SCO), both under convex and non-convex settings. For SO problems, we first analyze the generalization error of the STORM algorithm as a foundational step. We then extend our analysis to SCO problems by introducing an algorithmic framework that encompasses several popular algorithmic approaches. Through this framework, we conduct a generalization analysis, uncovering new insights into the impact of estimators on generalization. Subsequently, we provide a detailed analysis of three specific algorithms within this framework: SCGD, SCSC, and COVER, to explore the effects of different estimator strategies. Furthermore, in the context of SCO, we propose a novel definition of stability and a new decomposition of excess risk in the non-convex setting. Our analysis indicates two key findings: (1) In SCO problems, eliminating the estimator for the gradient of the inner function does not impact generalization performance while significantly reducing computational and storage overhead. (2) Faster convergence rates are consistently associated with better generalization performance.

1 Introduction

Recently, the Stochastic Compositional Optimization (SCO) problem has found extensive applications in machine learning, including model-agnostic meta-learning (MAML) [Finn *et al.*, 2017] and reinforcement learning [Dann *et al.*, 2014]. The SCO problem [Qi *et al.*, 2021; Chen *et al.*, 2021; Wang *et al.*, 2017; Gao and Huang, 2021] is formulated as follows:

$$\min_{x\in\mathcal{X}}\{F(x) = f \circ g(x) = \mathbb{E}_\nu[f_\nu(\mathbb{E}_\omega[g_\omega(x)])]\}, \quad (1)$$

where $f : \mathbb{R}^{d_1} \to \mathbb{R}^{d_2}$ and $g : \mathbb{R}^d \to \mathbb{R}^{d_1}$ are defined on a convex domain $\mathcal{X} \subset \mathbb{R}^d$, ν and ω are independent random variables. Here, $g_\omega(\cdot)$ and $f_\nu(\cdot)$ represent random functions parameterized by ω and ν, respectively. Solving the SCO problem is challenging because, in SCO, acquiring unbiased gradient estimates of the compositional function $F(x)$ is particularly difficult. In other words, the expectation $\mathbb{E}_{\nu,\omega}[\nabla f_\nu(g_\omega(x))]$ does not equal $\nabla f_\nu(g_\omega(x))$, making it infeasible to obtain an unbiased estimate of $F(x)$.

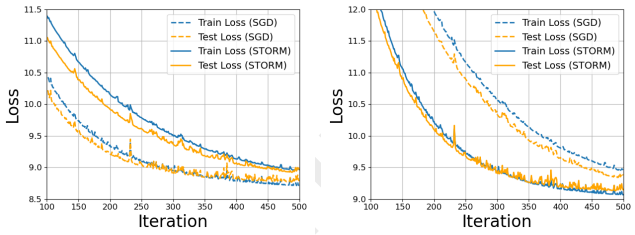
Due to the prevalence of SCO, many studies have been developed to address its challenges [Jiang *et al.*, 2022; Liu *et al.*, 2024]. Notable contributions include the development of Stochastic Composite Gradient Descent (SCGD) [Wang *et al.*, 2017], which is based on the momentum technique to estimate the inner function value and achieves an $O(T^{-1/4})$ convergence rate in non-convex settings. Additionally, Variance Reduction (VR) techniques [Johnson and Zhang, 2013; Fang *et al.*, 2018; Cutkosky and Orabona, 2019] have been developed. VR techniques typically employ a gradient estimator to track the gradient more accurately, updating it using estimated gradient values rather than relying solely on stochastic gradients. Building on the Variance Reduction (VR) technique, SCSC and COVER were proposed in [Chen *et al.*, 2021] and [Qi *et al.*, 2021], respectively. SCSC achieves a convergence rate of $O(T^{-1/4})$, while COVER improves this rate to $O(T^{-1/3})$ in non-convex settings.

Although numerous algorithms for solving the SCO problem have been developed based on different techniques for designing estimators, as shown above, much of the focus over the past few decades has been on improving convergence rates by using different types and quantities of estimators. However, the impact of estimator variations on the generalization of these algorithms is often overlooked. This aspect is crucial, as generalization serves as a key indicator of how well a learned model, trained on given training samples, performs on unseen test data [Bassily *et al.*, 2020; London *et al.*, 2016].

To explain the impact of estimators, we begin with the Stochastic Optimization (SO) problem [Zhang, 2004; Bottou *et al.*, 2018], which can be viewed as a special case of the SCO problem (when $g(x) = x$). It can be formulated as follows:

$$\min_{x\in\mathcal{X}}\{F(x) = \mathbb{E}_\nu[f_\nu(x)]\}, \quad (2)$$

*Corresponding Authors.



(a) Convex SO. (b) Non-convex SO.

Figure 1: Performance of SGD and STORM in both convex and non-convex settings.

Due to the complexity of the SCO problem, different algorithms choose different update strategies for different levels of the function, making it difficult to control variables for comparison. In contrast, the SO problem allows us to compare two algorithms—one with estimators and one without—making the comparison more intuitive and convenient. Therefore, we compare SGD with other variance-reduced (VR) algorithms within the SO problem. Among VR-based algorithms, STOchastic Recursive Momentum (STORM) [Cutkosky and Orabona, 2019] stands out, as it significantly improves the convergence rate from SGD's $O(T^{-1/4})$ to $O(T^{-1/3})$ in non-convex settings, bringing the solution closer to near-optimality.

We conducted a toy experiment to compare SGD and STORM in convex and non-convex settings, as shown in Figure 1. In the non-convex setting, STORM demonstrated a faster convergence rate than SGD, as shown in Figure 1b, and it also exhibited superior generalization performance because the gap between training loss and testing loss is smaller. However, in the convex setting (Figure 1a), the trend reversed. It can be noticed that the estimator not only changes the convergence rate of the algorithm, but also the generalization of the algorithm. This observation raises an interesting question: **How does the estimator affect the generalization?**

Therefore, in this paper, we analyze the effect of estimators on generalization under the SO and SCO problems. We first conduct a generalization analysis of STORM, comparing the results to SGD and exploring the estimator's impact on generalization as an initial step. Considering the complexity of the SCO problem, we then propose a generalized algorithmic framework that encompasses many existing algorithms and applies to multiple estimators and various estimation strategies—i.e., the objects to which the estimator is applied. Subsequently, we analyze the generalization of this framework in both convex and non-convex settings. Through our analysis, we identify the estimators that are crucial and those that are less important. Furthermore, we perform a generalization analysis of three representative algorithms—SCGD, SCSC, and COVER—under the SCO problem to compare how different estimation strategies impact algorithm generalization. Based on this analysis, we also establish a relationship between the convergence rate and generalization, summarized in Table 2. In summary, the main contributions of this paper are as follows:

- We propose a generalized algorithmic framework for the SCO problem. We then analyze the proposed framework in both convex and non-convex settings, along with the generalization performance of three representative algorithms—SCGD, SCSC, and COVER—to understand how different strategies influence generalization outcomes. In addition, we have found that convergence and generalization are closely related; specifically, a faster convergence rate tends to enhance generalization performance.
- For the non-convex setting of the SCO problem, we introduce a novel definition of stability. This new stability definition is based on the gradient of the objective function, recognizing that in a non-convex setting, the goal is often to find a stationary point rather than the global minimum. We also introduce a new excess risk decomposition for the SCO problem in the non-convex setting. To the best of our knowledge, this new definition and the new excess risk decomposition allow us, for the first time, to generalize our analysis of algorithms for the SCO problem in the non-convex setting. Through our analysis, we provide guidance on the use of estimators when designing algorithms.

2 Related Work

SO and SCO Algorithms. In addition to SGD [Zhang, 2004; Bottou *et al.*, 2018] and VR-based methods [Johnson and Zhang, 2013; Fang *et al.*, 2018; Cutkosky and Orabona, 2019] for stochastic optimization, several improved algorithms have emerged from various perspectives. For example, Nesterov acceleration [Nesterov, 1983; Attouch and Peypouquet, 2016] and adaptive learning rates [Duchi *et al.*, 2011; Kingma and Ba, 2014; Zhou *et al.*, 2018] can achieve an impressive $O(T^{-1/2})$ convergence rate. Due to biased gradient estimations on both inner and outer functions, the predominant trend in addressing SCO problems is leveraging VR techniques. For example, variants of SARAH [Nguyen *et al.*, 2017] and SPIDER [Fang *et al.*, 2018] achieve an impressive $O(T^{-1/3})$ convergence rate with large batch sizes [Zhang and Xiao, 2019]. COVER [Qi *et al.*, 2021] proposes a batch-free algorithm based on STORM with the same rate without using mini-batches. Furthermore, recent advancements extend VR techniques to federated SCO [Gao and Huang, 2021], multi-level SCO [Jiang *et al.*, 2022], and compositional minimax problems [Liu *et al.*, 2024]. Despite extensive exploration of these problems, their generalization remains under-explored.

Generalization Analysis. Algorithmic stability is a cornerstone concept in learning theory, gauging an algorithm's resilience to perturbations in the training dataset, closely intertwined with its learnability [Rakhlin *et al.*, 2005; Shalev-Shwartz *et al.*, 2010]. One of the most widely employed stability concepts is uniform stability [Bousquet and Elisseeff, 2002], often signaling nearly optimal generalization bounds with high probability [Feldman and Vondrak, 2019; Bousquet and Elisseeff, 2002]. While ML methods can excel in training data but falter in generalization, the uniform convergence approach in generalization analysis helps illuminate the discrepancy between training and testing across the entire hypothesis space [Foster *et al.*, 2018; Nagarajan and Kolter, 2019]. Initially, uniform convergence focused on function values [Bartlett and Mendelson, 2002], which is

less suited for stochastic optimization with nonconvex loss functions. Subsequent studies have improved upon this, encompassing uniform convergence of function values [Lei *et al.*, 2021], gradient risks in smooth problems [Ghadimi and Lan, 2013], and Moreau envelope gradients in weakly convex problems [Davis and Drusvyatskiy, 2019].

3 The Target of Analysis and Our Framework

In this section, we illustrate the target of the generalization analysis for both convex and non-convex settings, and then we introduce our proposed framework, followed by some necessary notation.

3.1 The Target of Analysis

We first introduce some notations to clarify our final target. Let $x^* = \arg\min_{x\in\mathcal{X}} F(x)$ be the model with the minimal population risk in $\mathcal{X}$. Let A be a randomized learning algorithm and $A(S)$ be the output model when applying A to the dataset S. Let $\|\cdot\|$ denote the Euclidean norm, and let $\nabla f(x)$ represent a subgradient of f at x. If f is differentiable, then $\nabla f(x)$ corresponds to the gradient of f at x. It is important to note that we use the same symbols, $F_S(x), F(x)$, to represent the empirical risk and population risk for both SO and SCO problems. However, their meanings differ in these two contexts, and we will provide detailed notations in Section 4 and 5 to clarify them in the following section.

The Convex Setting. For the convex setting that is lower bounded, there is always an optimal global minimum, denoted as x^*. The model's behavior is quantified by the population risk, with the goal of analyzing the excess risk $\mathbb{E}_{S,A}[F(A(S))] - F(x^*)$. The standard approach [Bousquet and Elisseeff, 2002] decomposes excess risk into two error terms as follows:

$$\mathbb{E}_{S,A}\big[F(A(S))\big] - F(x^*) = \mathbb{E}_{S,A}\big[F(A(S)) - F_S(A(S))\big] + \mathbb{E}_{S,A}\big[F_S(A(S)) - F_S(x^*)\big], \quad (3)$$

where we use the relation $\mathbb{E}_{S,A}[F_S(x^*)] = F(x^*)$ because x^* is independent of both A and S. We refer to the difference $F(A(S)) - F_S(A(S))$ mentioned in (3) as the generalization error, as it reflects how the model generalizes from training to testing behaviors. The term $F_S(A(S)) - F_S(x^*)$ is identified as the optimization error since it measures the algorithm's effectiveness in minimizing the empirical risk, which is widely used in [Hardt *et al.*, 2016; Kuzborskij and Lampert, 2018; Charles and Papailiopoulos, 2018].

The Non-Convex Setting. However, for non-convex learning problems, we focus on whether the learning algorithm can find an approximate stationary point, that is, $\|\nabla F_S(A(S))\| \leq \epsilon$ because it is difficult to find the global optimum [Zhang, 2004; Bottou *et al.*, 2018; Cutkosky and Orabona, 2019]. Consequently, the excess risk used in convex settings are not applicable. Instead, we employ the population gradient norm as the performance measure, shifting our final target to $\mathbb{E}_{S,A}[\|\nabla F(A(S))\|]$. Based on this measurement method, we have the following decomposition:

$$\mathbb{E}_{S,A}\big[\|\nabla F(A(S))\|\big] \leq \mathbb{E}_{S,A}\big[\|\nabla F(A(S)) - \nabla F_S(A(S))\|\big] + \mathbb{E}_{S,A}\big[\|\nabla F_S(A(S))\|\big]. \quad (4)$$

Similarly to the decomposition in the convex setting, we refer to the first term, $\mathbb{E}_{S,A}\big[\|\nabla F(A(S)) - \nabla F_S(A(S))\|\big]$ as the generalization error and to the second term, $\mathbb{E}_{S,A}\big[\|\nabla F_S(A(S))\|\big]$, as the optimization error.

3.2 The Proposed Framework

We first introduce some notations. Similar to [Wang *et al.*, 2017; Qu *et al.*, 2023; Yang *et al.*, 2023], we concern the case that the random variables ν and ω are independent. In practice, we do not know the population distributions for ν and ω for SCO problem but only have access to a set of training data $S = \{\nu_1, \ldots, \nu_n, \omega_1, \ldots, \omega_m\}$. Now we give the framework for solving (2). For the algorithms of the SCO problem, their main difference lies in the different designs of u_t and v_t. Some commonly used approaches for estimating the inner function value are as follows:

Algorithm 1 Framework of SCO

Inputs: Training data $S = \{\nu_1, \ldots, \nu_n, \omega_1, \ldots, \omega_m\}$; Number of iterations T, parameter sequence $\{\eta_t\}$, $\{\beta_t\}$
Initialize $x_0 \in \mathcal{X}$ and $y_0 \in \mathbb{R}^d$
for $t = 0$ to $T-1$ **do**
  Randomly sample $j_t \in [1, m]$, obtain $g_{\omega_{j_t}}(x_t)$ and $\nabla g_{\omega_{j_t}}(x_t) \in \mathbb{R}^{d\times d_2}$
  Estimate inner function value u_t according to Eq.(5)
  Estimate inner function gradient v_t according to Eq. (6) or Eq. (7)
  Randomly sample $i_t \in [1, n]$, obtain $\nabla f_{\nu_{i_t}}(u_t) \in \mathbb{R}^d$
  Calculate the total gradient $\mathbf{v}_t = v_t \cdot \nabla f_{\nu_{i_t}}(u_t)$
  Update:
  $x_{t+1} = \Pi_{\mathcal{X}}\big(x_t - \eta_t \mathbf{v}_t\big)$
end for
Outputs: $A(S) = x_T$ or $x_\tau \sim \texttt{Unif}(\{x_t\}_{t=1}^T)$

$$\begin{aligned} &\text{Momentum-type}: u_t = (1-\beta_t)u_{t-1} + \beta_t g_{\omega_{j_t}}(x_t), \\ &\text{VR-type}: u_t = (1-\beta_t)u_{t-1} + \beta_t g_{\omega_{j_t}}(x_t) \\ &\qquad + (1-\beta_t)(g_{\omega_{j_t}}(x_t) - g_{\omega_{j_t}}(x_{t-1})). \end{aligned} \quad (5)$$

For the inner function gradient, there are typically two cases. The first is to use the stochastic gradient directly, i.e.,

$$\text{Vanilla-type:} \quad v_t = \nabla g_{\omega_{j_t}}(x_t). \quad (6)$$

In other cases, an estimator is used instead of the stochastic gradient for updating. There are many types of estimators, and we list a few common ones below:

$$\begin{aligned} &\text{Momentum-type:} \quad v_t = (1-\beta_t)v_{t-1} + \beta_t \nabla g_{\omega_{j_t}}(x_t), \\ &\text{VR-type:} \quad v_t = (1-\beta_t)v_{t-1} + \beta_t \nabla g_{\omega_{j_t}}(x_t) \\ &\qquad + (1-\beta_t)(\nabla g_{\omega_{j_t}}(x_t) - \nabla g_{\omega_{j_t}}(x_{t-1})). \end{aligned} \quad (7)$$

It is worth noting that there are many other types of methods, such as SPIDER [Fang *et al.*, 2018], SVRG [Johnson and Zhang, 2013], and SAGA [Defazio *et al.*, 2014]. It is also important to emphasize that algorithms based on the proposed framework are not limited to the aforementioned three

methods but also include others, such as NASA [Ghadimi *et al.*, 2020], CIVR [Zhang and Xiao, 2019], and Compositional SVRG [Lian *et al.*, 2017], among others. Naturally, these algorithms select different parameter sequences depending on the estimator used.

Subsequently, we aim to analyze the generalization of the proposed framework. To achieve this, we derive generalization bounds with the help of stability analysis. Therefore, we first introduce the definition of stability. We begin by presenting the definition of stability in the context of the SO problem, followed by our newly introduced definition of stability for the SCO problem.

3.3 Definition of Stability

Definition 1 (Uniform Stability for SO)**.** *Let A be a randomized algorithm. Let $S = (\nu_1, \nu_2, \ldots, \nu_n)$ be drawn i.i.d from $\mathcal{D}$. Let S and S' be neighboring datasets, differing by at most one data point.*

- *We say A is ϵ-uniformly-stable in function values if for all neighboring datasets S, S', such that $\sup_\nu \mathbb{E}_A\big[f_\nu(A(S)) - f_\nu(A(S'))\big] \leq \epsilon$.*
- *We say A is ϵ-uniformly-stable in gradients if for all neighboring datasets S, S', such that $\sup_\nu \mathbb{E}_A\big[\|\nabla f_\nu(A(S)) - \nabla f_\nu(A(S'))\|^2\big] \leq \epsilon^2$.*

Definition 1 defines stability for convex and non-convex settings. In convex settings, the goal is to find the global minimum, so stability is defined by function values. In non-convex settings, the focus is on finding an approximate stable point [Cutkosky and Orabona, 2019; Levy *et al.*, 2021; Lei, 2023], where the gradient approaches zero, leading to stability being defined by the function gradient.

When considering SCO problems in (1), the definition of stability becomes more complex. Changes in the sample may occur in either the set $\{\nu_1, \nu_2, \ldots, \nu_n\}$ or the set $\{\omega_1, \omega_2, \ldots, \omega_m\}$. Although various stability concepts, such as locally elastic stability [Deng *et al.*, 2021; Zhang *et al.*, 2021; Qu *et al.*, 2022], hypothesis stability [Charles and Papailiopoulos, 2018; Bousquet *et al.*, 2020], and PAC-Bayesian stability [Li *et al.*, 2019; Rivasplata *et al.*, 2020], have been developed, they are not suited for non-convex SCO problems. To address this, we introduce a novel stability concept designed for this context and utilize it in our analysis. Let $S^{i,\nu}$ denote the change at the i-th data point in $\{\nu_1, \nu_2, \ldots, \nu_n\}$, and $S^{j,\omega}$ denote the change at the j-th data point in $\{\omega_1, \omega_2, \ldots, \omega_m\}$, we propose a new definition of stability.

Definition 2 (Uniform Stability for SCO)**.** *Let A be a randomized algorithm.*

- *We say A is ϵ-uniformly-stable in function values if for all neighboring datasets $S, S^{i,\nu}$, we have $\sup_{\nu,\omega} \mathbb{E}_A\big[f_\nu(g_\omega(A(S))) - f_\nu(g_\omega(A(S^{i,\nu})))\big] \leq \epsilon_\nu$, and for all neighboring datasets $S, S^{j,\omega}$, we have $\sup_\omega \mathbb{E}_A\big[g_\omega(A(S)) - g_\omega(A(S^{j,\omega}))\big] \leq \epsilon_\omega$.*
- *We say A is ϵ-uniformly-stable in gradients if for all neighboring datasets $S, S^{i,\nu}$, we have $\sup_{\nu,\omega} \mathbb{E}_A\big[\|\nabla f_\nu(g_\omega(A(S))) - \nabla f_\nu(g_\omega(A(S^{i,\nu})))\|^2\big] \leq \epsilon_\nu^2$, and for all neighboring datasets $S, S^{j,\omega}$, we have $\sup_\omega \mathbb{E}_A\big[g_\omega(A(S)) - g_\omega(A(S^{j,\omega}))\big] \leq \epsilon_\omega$.*

In Definition 2, we stabilize the function value of the inner function $g(\cdot)$ for both settings. An alternative stability definition considers: $\sup_\omega \mathbb{E}_A[\|\nabla g_\omega(A(S)) - \nabla g_\omega(A(S^{j,\omega}))\|^2] \leq \epsilon_\omega^2$ for the non-convex setting. However, this approach relies on the chain rule and requires additional assumptions, specifically that the gradient of the outer function is bounded, i.e. $\|\nabla f_\nu(\cdot)\|$ is bounded, to control $\|\nabla f_{\nu_i}(\frac{1}{m}\sum_{j=1}^m g_{\omega_j}(A(S))) - \nabla f_{\nu_{i_t}}(g(A(S)))\|$. In contrast, Definition 2 leverages the Lipschitz continuity of the outer function, avoiding the need for gradient boundedness.

Our final target is presented in (3) and (4). With the concept of stability and the stability bounds of algorithms, we can derive the generalization error, which corresponds to the first term on the right side of (3) and (4). By also considering the optimization error, represented by the second term, we can achieve the desired results.

4 Theoretical Analysis of SOs

For the SO problem in (2), the training and testing behavior can be typically measured by the empirical risk $F_S(x) := \frac{1}{n}\sum_{i=1}^n f_{\nu_i}(x)$ and the population risk $F(x) := \mathbb{E}_\nu[f_\nu(x)]$. To facilitate our proof, we first present the following essential definitions. These concepts are fundamental to the generalization analysis of SO problems [Hardt *et al.*, 2016; Bousquet and Elisseeff, 2002; Lei, 2023].

Definition 3. *Let $f : \mathbb{R}^d \to \mathbb{R}$. There exist constants $L_f, C_f > 0$ holding the following conditions:*

- *We say f is Lipschitz continuous if $\sup_\nu \|f_\nu(x) - f_\nu(\hat{x})\| \leq L_f\|x - \hat{x}\|, \forall x, \hat{x} \in \mathbb{R}^d$.*
- *We say f is smoothness if $\sup_\nu \|\nabla f_\nu(x) - \nabla f_\nu(\hat{x})\| \leq C_f\|x - \hat{x}\|, \forall x, \hat{x} \in \mathbb{R}^d$.*

Definition 4. *Let $f : \mathbb{R}^d \to \mathbb{R}$ and the constant $\sigma_g > 0$. With probability 1 w.r.t S, it holds that $\sup_{x\in\mathcal{X}} \frac{1}{n}\sum_{i=1}^n [\|\nabla f_{\nu_i}(x) - \nabla f_S(x)\|^2] \leq \sigma_g^2$.*

As a preliminary step, and in conjunction with above two definitions, we employ the stability concept outlined in Definition 1 to derive the results of the SO problem.

Theorem 1 (Convex of STORM)**.** *Let F_S be convex, $\mathbb{E}_A[\|x_t - x_*^S\|] \leq D_x$ with $t > 0$, $\sup_\nu \mathbb{E}_A[f_\nu(A(S))] \leq L$, $\forall S$ and $\mathbb{E}_A[\|v_0 - \nabla f_S(x_0)\|^2] \leq \Delta_v$. If we choose $\beta \geq 8L_f^2\eta^2$, for any $c > 0$, $\mathbb{E}_{S,A}[F(A(S))] - F(x^*)$ satisfies:*

$$O\bigg(\Big(\eta + \frac{1}{\sqrt{\eta}}\Big)\Big(\frac{\Delta_v}{(T\beta)^c} + \beta\sigma_g^2 + \frac{L_f^4\eta^2}{\beta}\Big) + \frac{D_x}{\eta T} + D_x\sqrt{\eta} + \eta L_f^2 + \frac{LT}{n}\bigg).$$

When we choose that $\eta \asymp T^{-2/3}$, $\beta \asymp T^{-2/3}$, and $T \asymp L^{-3/4}n^{3/4}$, then we can obtain that $\mathbb{E}_{S,A}\big[F(A(S))\big] - F(x^*) = O(n^{-1/4}L^{1/4})$.

Remark 1. Theorem 1 shows that SGD requires more iterations, $T \asymp n$, to achieve a better generalization result of $O(n^{-1/2})$. This makes direct comparison with STORM challenging under different iteration regimes. To ensure fairness, we fix the iterations to $T \asymp n^{3/4}$for both methods. Under this condition, SGD achieves an excess error bound of $O(n^{-3/8})$, while STORM attains $O(n^{-1/4})$. Clearly, SGD exhibits better generalization performance in this scenario. This explains STORM's inferior performance compared to SGD in the convex toy example (Figure 1a).

We now examine STORM in non-convex settings.

Theorem 2 (Non-convex of STORM)**.** *If F_S is non-convex, $\eta_t = O(T^{-1/3}), \beta_t = O(T^{-2/3})$, $\sup_\nu \mathbb{E}_A[\|\nabla f_\nu(A(S))\|^2] \leq G^2$, $\forall S$ and $\mathbb{E}_A[\|v_0 - \nabla f_S(x_0)\|^2] \leq \Delta_v$, the output $A(S) = x_T$ satisfies:*

$$\mathbb{E}_{S,A}[\|\nabla F(A(S))\|] = O\Big(\sqrt{\eta_T T} + T^{-1/3} + \sqrt{\frac{G^2 T}{n}}\Big).$$

If $T \asymp G^{-6/5} n^{3/5}$, we can obtain that $\mathbb{E}_{S,A}[F(A(S))] - F(x^*) = O(G^{2/5} n^{-1/5})$.

In the non-convex setting, the proof reveals that generalization heavily depends on the estimator's error. A detailed analysis of the estimator's impact will be discussed later in the SCO problem, which includes the SO problem.

It important to notice that both Theorems 1-2 depend on the assumptions that function values and gradients are bounded when the algorithm's final output is reused as input. These widely used assumptions [Hardt *et al.*, 2016; Lei, 2023; Wang *et al.*, 2024] are automatically met by applying a projection operator to $A(S)$, thus avoiding iterative projections.

In the following discussion, we will examine the SCO problem and conduct a comparative analysis of the framework and three algorithms across both settings, focusing on the impact of the estimator on generalization.

5 Theoretical Analysis of SCO

When the problem involves SCO [Qi *et al.*, 2021; Chen *et al.*, 2021; Wang *et al.*, 2017; Gao and Huang, 2021], the objective function $f(x)$ is extended into the compositional function $f(g(x))$. The empirical risk for SCO problems is defined as follows: $\min_{x\in\mathcal{X}} F_S(x) := \frac{1}{n}\sum_{i=1}^n f_{\nu_i}(\frac{1}{m}\sum_{j=1}^m g_{\omega_j}(x))$, and the population risk as $F(x) := \mathbb{E}_\nu[f_\nu(\mathbb{E}_\omega[g_\omega(x)])]$. Due to the difference, we need to re-define the corresponding Lipschitz continuity and smoothness for the SCO problem.

Definition 5. *Let $f : \mathbb{R}^{d_1} \to \mathbb{R}^{d_2}$ and $g : \mathbb{R}^d \to \mathbb{R}^{d_1}$. There exist constants $L_f, L_g, C_f > 0$ holding the following conditions:*

- *We say f is Lipschitz continuous, if $\sup_\nu \|f_\nu(y) - f_\nu(\hat{y})\| \leq L_f\|y - \hat{y}\|$ for all $y, \hat{y} \in \mathbb{R}^{d_1}$.*
- *We say f is smoothness, if $\sup_\nu \|\nabla f_\nu(y) - \nabla f_\nu(\hat{y})\| \leq C_f\|y - \hat{y}\|$ for all $y, \hat{y} \in \mathbb{R}^{d_1}$.*
- *We say g is Lipschitz continuous, if $\sup_\omega \|g_\omega(x) - g_\omega(\hat{x})\| \leq L_g\|x - \hat{x}\|$, $\forall x, \hat{x} \in \mathbb{R}^d$.*
- *We say g is smoothness, if $\sup_\omega \|\nabla g_\omega(x) - \nabla g_\omega(\hat{x})\| \leq C_g\|x - \hat{x}\|$ for all $x, \hat{x} \in \mathbb{R}^d$.*

Definition 6. *Let $f : \mathbb{R}^{d_1} \to \mathbb{R}^{d_2}$ and $g : \mathbb{R}^d \to \mathbb{R}^{d_1}$. There exist constants $\sigma_g, \sigma_{g'} > 0$ holding that With probability 1 w.r.t S, it holds that $\sup_{x\in\mathcal{X}} \frac{1}{m}\sum_{j=1}^m [\|g_{\omega_j}(x) - g_S(x)\|^2] \leq \sigma_g^2$ and $\sup_{x\in\mathcal{X}} \frac{1}{m}\sum_{j=1}^m [\|\nabla g_{\omega_j}(x) - \nabla g_S(x)\|^2] \leq \sigma_{g'}^2$.*

With the above definitions, we can then establish the quantitative relationship between Uniform Stability (i.e., Definition 2) and generalization error.

Theorem 3 (Generalization via Stability in Gradients)**.** *Let A be ϵ-uniformly-stable in gradients. Assume for any ω and ν, the function $g_\omega(\cdot)$ and $f_\nu(\cdot)$ is differentiable. Then,*

$$\begin{aligned}&\mathbb{E}_{S,A}[\|\nabla F(A(S)) - \nabla F_S(A(S))\|] \\ &\leq 4\epsilon_\nu + 4C_f\epsilon_\omega + \sqrt{n^{-1}\mathbb{E}_{S,A}[\mathbb{V}_\nu(\nabla f_\nu(g(A(S))))]} \\ &\quad + C_f\sqrt{m^{-1}\mathbb{E}_{S,A}[\mathbb{V}_\omega(g_\omega(A(S)))]}.\end{aligned}$$

Remark 2. Deriving the relationship between stability and generalization for SCO algorithms in non-convex settings presents significant technical challenges. Specifically, the first term in the convex setting decomposition is $\mathbb{E}_{S,A}[\mathbb{E}_\nu[f_\nu(g(A(S)))] - \frac{1}{n}\sum_{i=1}^n f_{\nu_i}(g(A(S)))]$, as shown in [Yang *et al.*, 2023]. It can be handled similarly to the non-compositional setting: $\mathbb{E}_{S,A}\left[\mathbb{E}_\nu\left[f_\nu(g(A(S)))\right] - \frac{1}{n}\sum_{i=1}^n f_{\nu_i}(g(A(S)))\right] = \mathbb{E}_{S,A,S',\nu}\left[\frac{1}{n}\sum_{i=1}^n \left(f_{\nu_i}\left(g\left(A\left(S^{i,\nu}\right)\right)\right) - f_{\nu_i}(g(A(S)))\right] \leq L_f\left\|g\left(A\left(S^{i,\nu}\right)\right) - g(A(S))\right\|$. However, the corresponding term in the non-convex setting is: $2\mathbb{E}_{S,A}[\|\mathbb{E}_\nu[\nabla f_\nu(g(A(S)))] - \frac{1}{n}\sum_{i=1}^n \nabla f_{\nu_i}(g(A(S))\|^2]$. Since the Euclidean norm and expectation operations cannot be interchanged, the method for the convex setting [Yang *et al.*, 2023] cannot be applied. We address this by re-decomposing the first term, inspired by [Hardt *et al.*, 2016].

We now consider a class of algorithms, specifically SCO sampling-determined algorithms. Details on sampling-determined algorithms can be found in Theorem 7 in the Appendix.

The generalization performance of the SCO framework is formalized in the following theorem, with the outcomes for the three algorithms summarized in Table 2.

Theorem 4 (Convex of SCO Framework)**.** *Let F_S be convex, $\mathbb{E}_A[\|x_t - x_*^S\|] \leq D_x$ for any $t > 0$, $\sup_\omega \mathbb{E}_A[g_\omega(A(S))] \leq L_\omega$ and $\sup_{\nu,\omega} \mathbb{E}_A[f_\nu(g_\omega(A(S)))] \leq L_\nu$, $\forall S$, for any constant $\gamma > 0$, define $\epsilon_t^{g_S} := \mathbb{E}_A[\|g_S(x_t) - u_t\|^2]$, if the framework is updated using Eq.(6), $\mathbb{E}_A[F_S(A(S)) - F_S(x_*^S)]$ satisfies:*

$$O\Big(\frac{D_x^2}{\eta T} + \eta + \frac{D_x^2}{\gamma} + \frac{\gamma}{T}\sum_{t=1}^T \epsilon_t^{g_S} + \frac{L_\nu T}{n} + \frac{L_f L_\omega T}{m}\Big).$$

If the framework is updated using Eq.(7), $\mathbb{E}_A[F_S(A(S)) - F_S(x_*^S)]$ satisfies:

$$\begin{aligned}O\Big(&\frac{D_x^2}{\eta T} + \eta + \frac{D_x^2}{\gamma} + \frac{\gamma}{T}\sum_{t=1}^T \epsilon_t^{g_S} + \frac{L_\nu T}{n} + \frac{L_f L_\omega T}{m} \\ &+ \frac{\gamma+\eta}{T}\sum_{t=1}^T \mathbb{E}[\|v_t - \nabla g_S(x_t)\|^2]\Big).\end{aligned}$$

Method	Setting	Excess Risk	β	η
SCGD	Convex	$D_x^2(\eta T)^{-1} + \eta + \eta\beta^{-1} + (\beta^2 + D_x^2\beta)\eta^{-1} + L_\nu T n^{-1} + L_\omega T m^{-1}$	$O\left(T^{-3/4}\right)$	$O\left(T^{-1/2}\right)$
	Non-Convex	$(\eta_T T)^{-1/2} + (L_f^2 L_g + \sigma_g^2)T^{-1/4} + T^{1/2} G_\nu n^{-1/2} + T C_f L_\omega m^{-1}$	$O\left(T^{-3/4}\right)$	$O\left(T^{-1/2}\right)$
SCSC	Convex	$D_x^2(\eta T)^{-1} + \eta + \eta^{\frac{3}{2}}\beta^{-1} + \beta\eta^{-1/2} + D_x^2\eta^{1/2} + L_\nu T n^{-1} + L_\omega T m^{-1}$	$O\left(T^{-2/3}\right)$	$O\left(T^{-2/3}\right)$
	Non-Convex	$(\eta_T T)^{-1/2} + (L_f^2 L_g^4 + \sigma_g^2 + \frac{1}{2}L_g^2 L_f^2)T^{-1/4} + T^{1/2} G_\nu n^{-1/2} + T C_f L_\omega m^{-1}$	$O\left(T^{-1/3}\right)$	$O\left(T^{-2/3}\right)$
COVER	Convex	$D_x^2(\eta T)^{-1} + \eta + D_x^2\eta^{1/2} + \beta\eta + \eta^3\beta^{-1} + (\eta^{-1/2} + D_x^2)(\beta + \eta^2\beta^{-1}) + L_\nu T n^{-1} + L_\omega T m^{-1}$	$O\left(T^{-2/3}\right)$	$O\left(T^{-2/3}\right)$
	Non-Convex	$(\eta_T T)^{-1/2} + T^{-1/3} + T^{1/2} G_\nu n^{-1/2} + T C_f L_\omega m^{-1}$	$O\left(T^{-1/3}\right)$	$O\left(T^{-2/3}\right)$

Table 1: Summary of the excess risk of the three algorithms, where β and η represent the recommended values for the parameters associated with the algorithm's optimal excess risk.

Problem	VR	Method	Setting	Reference	Convergence	Iteration	Bound
SO	✗	SGD	Convex	[Hardt *et al.*, 2016]	$O(T^{-1/2})$	$T \asymp n$	$n^{-1/2}$
			Non-Convex	[Lei, 2023]	$O(T^{-1/4})$	$T \asymp n^{2/3}$	$n^{-1/6}$
	✓	STORM [Cutkosky and Orabona, 2019]	Convex	Ours	$O(T^{-1/3})$	$T \asymp n^{3/4}$	$n^{-1/4}$
			Non-Convex	Ours	$O(T^{-1/3})$	$T \asymp n^{3/5}$	$n^{-1/5}$
SCO	✗	SCGD [Wang *et al.*, 2017]	Convex	[Yang *et al.*, 2023]	$O(T^{-1/4})$	$T \asymp n^{7/2}$	$n^{-1/2}$
				Ours	$O(T^{-1/4})$	$T \asymp n^{4/5}$	$n^{-1/5}$
			Non-Convex	Ours	$O(T^{-1/4})$	$T \asymp n^{2/3}$	$n^{-1/6}$
	✓	SCSC [Chen *et al.*, 2021]	Convex	[Yang *et al.*, 2023]	$O(T^{-1/3})$	$T \asymp n^{5/2}$	$n^{-1/2}$
				Ours	$O(T^{-1/3})$	$T \asymp n^{3/4}$	$n^{-1/4}$
			Non-Convex	Ours	$O(T^{-1/4})$	$T \asymp n^{2/3}$	$n^{-1/6}$
	✓	COVER [Qi *et al.*, 2021]	Convex	Ours	$O(T^{-1/3})$	$T \asymp n^{3/4}$	$n^{-1/4}$
			Non-Convex	Ours	$O(T^{-1/3})$	$T \asymp n^{3/5}$	$n^{-1/5}$

Table 2: Summary of Theoretical Results: The risk bounds are optimized by selecting an optimal value for T iterations that balances the trade-off between generalization and optimization. Here, n represents the number of samples, with smaller bounds indicating better results.

Remark 3. According to Theorem 4, using an estimator for the gradient of the inner function in the convex setting does not yield significant improvements. Instead, the extra term $\frac{\gamma+\eta}{T}\sum_{t=1}^{T}\mathbb{E}[\|v_t - \nabla g_S(x_t)\|^2]$ suggests that such an estimator may lead to worse generalization. To further investigate the effect of the estimator on generalization, we compare the three algorithms. SCGD and SCSC employ different estimators. For SCGD, the corresponding term is $\frac{\eta}{\beta} + \frac{\beta^2 + D_x^2\beta}{\beta}$, while for SCSC, it is $\frac{\beta^{\frac{3}{2}}}{\beta} + \frac{\beta}{\sqrt{\eta}} + D_x^2\sqrt{\eta}$. Although SCSC and COVER use the same estimator, COVER additionally incorporates an estimator for the inner function values, resulting in extra terms such as $D_x^2(\beta + \frac{\eta^2}{\beta})$, as summarized in Table 1. This aligns with Theorem 4 and explains why COVER did not outperform SCSC in the convex setting.

Remark 4. Although [Yang *et al.*, 2023] established the excess risk for convex and strongly convex settings, it introduced additional constraints, such as $\eta \leq \min\{1/n, 1/m\}$, which may not always be feasible in practice. A very small learning rate is required for large datasets, potentially leading to slow convergence. In contrast, our analysis leverages sampling-determined algorithms to derive a more informative excess risk at a practical learning rate. Furthermore, we introduce COVER, an algorithm not considered in [Yang *et al.*, 2023]. This is significant because SCGD and SCSC differ only in their estimator techniques. Both algorithms estimate the inner function values, making it difficult to study the effect of the estimator on generalization in isolation.

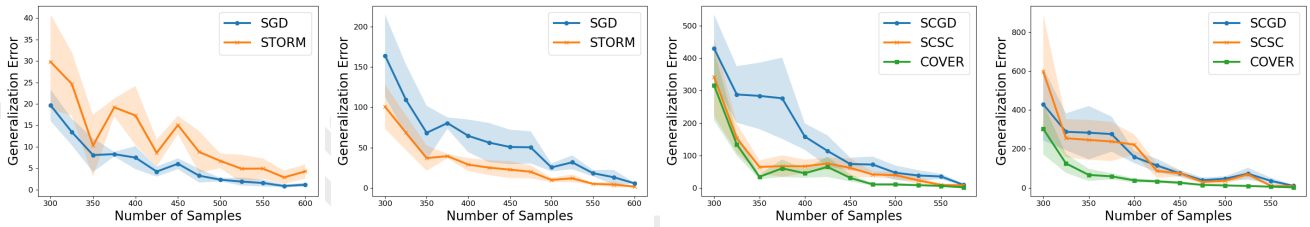
Accordingly, we will demonstrate the generalization performance of the framework in more complex and broader non-convex settings, as follows:

Theorem 5. *Let F_S be non-convex, and $\mathbb{E}_A[\|x_t - x_*^S\|] \leq D_x$ for any $t > 0$, $\sup_\omega \mathbb{E}_A[g_\omega(A(S))] \leq L_\omega$, and $\sup_{\nu,\omega}\mathbb{E}_A[f_\nu(g_\omega(A(S)))] \leq L_\nu$ for all S. For any $\mathcal{P}_t > 0$, define $\epsilon_t^{g_S} := \mathbb{E}_A[\|g_S(x_t) - u_t\|^2]$ and $\epsilon_t^{\nabla g_S} := \mathbb{E}_A[\|\nabla g_S(x_t) - v_t\|^2]$. Denote $\theta = 2L_f^2\sigma_g^2 + 2L_g^2C_f^2 - \frac{1}{4}L_f^2L_g^2$, if the framework is updated using Eq.(6), $\mathbb{E}_{S,A}[\|\nabla F(A(S))\|]$ satisfies:*

$$O\Big(\sqrt{\frac{\Delta_F}{\eta_T T}} + \sqrt{\frac{\mathcal{P}_{T+1}}{\eta_T T}\epsilon^{g_S}} + \sqrt{\frac{TG_\nu^2}{n}} + \frac{TC_f L_\omega}{m} + \sqrt{\frac{L_g^2 C_f^2}{\eta_T T}\sum_{t=1}^{T}\eta_t\epsilon^{g_S} + \theta\sum_{t=1}^{T}\eta_t}\Big),$$

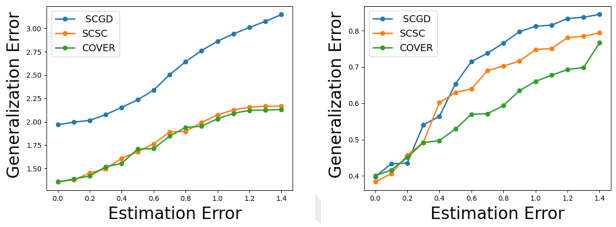
if the framework is updated using Eq.(7), $\mathbb{E}_{S,A}[\|\nabla F(A(S))\|]$ satisfies:

$$O\Big(\sqrt{\frac{\Delta_F}{\eta_T T}} + \sqrt{\frac{\mathcal{P}_{T+1}}{\eta_T T}(\epsilon_{T+1}^{g_S} + \epsilon_{T+1}^{\nabla g_S})} + \sqrt{\frac{TG_\nu^2}{n}} + \frac{TC_f L_\omega}{m} + \frac{1}{\sqrt{\eta_T T}}\sqrt{\sum_{t=1}^{T}\eta_t\epsilon_t^{g_S} + \sum_{t=1}^{T}\eta_t\epsilon_t^{\nabla g_S} - \sum_{t=1}^{T}\eta_t\|v_t\|^2}\Big).$$



(a) Convex SO. (b) Non-convex SO. (c) Convex SCO. (d) Non-convex SCO.

Figure 2: Generalization error on SO and SCO problem under convex and non-convex settings.



(a) Convex SCO. (b) Non-convex SCO.

Figure 3: Generalization error varies with estimation error.

Remark 5. According to Theorem 5, the main difference between the two is that one has the extra term $\sqrt{\theta \sum_{t=1}^{T} \eta_t}$, while the other has $\sum_{t=1}^{T} \eta_t \epsilon_t^{\nabla g_S}$. In this case, it is difficult to directly determine which case is more desirable. However, by analyzing the three algorithms, we find that the error of the COVER algorithm's estimator is $O(T^{-1/3})$, while it is $O(T^{-1/4})$ for both SCGD and SCSC. We also examined other algorithms that employ the same strategy as SCGD and SCSC, i.e., without estimating the gradient of the inner function, and found that their estimator errors all reach $O(T^{-1/4})$. This suggests that estimating the gradient of the inner function in the non-convex setting further improves algorithm performance.

Remark 6. By comparing Table 2, we observe a positive correlation between algorithms' convergence rates and their generalization performance. Specifically, We can find that whether it is a SO problem or a SCO problem, and whether it is a convex setting or a non-convex setting, algorithms with faster convergence rates tend to exhibit superior generalization, both in convex and non-convex settings.

6 Performance Evaluation

We conduct simulations to validate our theoretical results by examining the relationship between convergence and generalization. To simulate high-dimensional issues, we set the dimension of $\mathcal{X}$ to 100. We generate data using the objective function and add Gaussian noise with mean 0 and variance 1 to each dimension to mimic stochastic optimization. We use MSE loss for the convex setting and combine the tanh activation function with MSE loss for the non-convex setting, resulting in a total loss of MSE loss $+ \alpha \cdot \tanh(y_{\text{predict}})$, where $\alpha = 0.1$ in both SO and SCO problems. For SO problems, we use linear regression in both convex and non-convex settings. For SCO problems, we generate two datasets, S_1 and S_2, with $|S_1| = |S_2|$. The inner function $g(\cdot)$ fits S_1, and $f(g(\cdot))$ fits S_2. We aim to minimize the overall loss of the fits on both datasets. For iteration counts of different methods, we use the results from Table 2 and round to the nearest integer.

For SO problems (Figures 2a and 2b), SGD outperforms STORM in convex settings with faster convergence and better generalization, while STORM excels in non-convex settings. In the convex SCO setting (Figure 2c), SCGD converges slower than SCSC and COVER, which perform similarly, with SCGD also showing higher generalization error. In the non-convex setting, COVER converges faster than SCGD and SCSC, with Figure 2d demonstrating superior generalization for COVER. Overall, generalization error is lower in convex settings for both SO and SCO, consistent with theory.

To further test the effect of estimator errors on generalization, we conducted another set of simulation experiments. Since each algorithm uses a different number of estimators—i.e., COVER uses estimators for both the inner function value and its gradient, while SCGD and SCSC only use estimators for the inner function value—we added extra noise only to the estimators common to SCGD, SCSC, and COVER (i.e., those for the inner function values) for fairness. From Figure 3, it can be observed that larger estimation errors lead to worse generalization performance.

7 Conclusion

In this paper, we systematically analyze the impact of estimators on generalization. We first analyze the generalization of STORM and compare it with SGD under the SO problem. Later on, we propose a general framework for SCO that incorporates many existing popular algorithms, and then generalize it for both convex and non-convex settings. To analyze the non-convex setting under SCO, we also introduce a new definition of excess risk decomposition and stability in gradients. Finally, we select three representative algorithms and perform an analysis on them to further explore the effect of different estimator strategies on generalization. Through our analysis, we establish guidelines for using estimators when designing algorithms for SCO. Specifically, in the convex setting, the desired generalization result can be guaranteed without employing any estimator for the gradient of the inner function, which reduces computational and storage overhead. Additionally, we identify a relationship between convergence and generalization: algorithms with faster convergence rates tend to exhibit better generalization performance.

Acknowledgments

This work was supported by National Natural Science Foundation of China (NSFC) (62302525), the Project of Xiangjiang Laboratory (24XJJCYJ01003), the NSFC (U24A20256, 62202293, 62303306), the Science and Technology Innovation Program of Hunan Province (2022RC1031), the Natural Science Foundation of Hunan Province (2024JJ6527, 2025JJ50374), and the High Performance Computing Center of Central South University.

References

[Attouch and Peypouquet, 2016] Hedy Attouch and Juan Peypouquet. The rate of convergence of nesterov's accelerated forward-backward method is actually faster than 1/kˆ2. *SIAM Journal on Optimization*, 26(3):1824–1834, 2016.

[Bartlett and Mendelson, 2002] Peter L Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3(Nov):463–482, 2002.

[Bassily *et al.*, 2020] Raef Bassily, Vitaly Feldman, Cristóbal Guzmán, and Kunal Talwar. Stability of stochastic gradient descent on nonsmooth convex losses. *Advances in Neural Information Processing Systems*, 33:4381–4391, 2020.

[Bottou *et al.*, 2018] Léon Bottou, Frank E Curtis, and Jorge Nocedal. Optimization methods for large-scale machine learning. *SIAM review*, 60(2):223–311, 2018.

[Bousquet and Elisseeff, 2002] Olivier Bousquet and André Elisseeff. Stability and generalization. *The Journal of Machine Learning Research*, 2:499–526, 2002.

[Bousquet *et al.*, 2020] Olivier Bousquet, Yegor Klochkov, and Nikita Zhivotovskiy. Sharper bounds for uniformly stable algorithms. In *Conference on Learning Theory*, pages 610–626. PMLR, 2020.

[Charles and Papailiopoulos, 2018] Zachary Charles and Dimitris Papailiopoulos. Stability and generalization of learning algorithms that converge to global optima. In *International conference on machine learning*, pages 745–754. PMLR, 2018.

[Chen *et al.*, 2021] Tianyi Chen, Yuejiao Sun, and Wotao Yin. Solving stochastic compositional optimization is nearly as easy as solving stochastic optimization. *IEEE Transactions on Signal Processing*, 69:4937–4948, 2021.

[Cutkosky and Orabona, 2019] Ashok Cutkosky and Francesco Orabona. Momentum-based variance reduction in non-convex sgd. *Advances in neural information processing systems*, 32, 2019.

[Dann *et al.*, 2014] Christoph Dann, Gerhard Neumann, and Jan Peters. Policy evaluation with temporal differences: A survey and comparison. *The Journal of Machine Learning Research*, 15(1):809–883, 2014.

[Davis and Drusvyatskiy, 2019] Damek Davis and Dmitriy Drusvyatskiy. Stochastic model-based minimization of weakly convex functions. *SIAM Journal on Optimization*, 29(1):207–239, 2019.

[Defazio *et al.*, 2014] Aaron Defazio, Francis Bach, and Simon Lacoste-Julien. Saga: A fast incremental gradient method with support for non-strongly convex composite objectives. *Advances in neural information processing systems*, 27, 2014.

[Deng *et al.*, 2021] Zhun Deng, Hangfeng He, and Weijie Su. Toward better generalization bounds with locally elastic stability. In *International Conference on Machine Learning*, pages 2590–2600. PMLR, 2021.

[Duchi *et al.*, 2011] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.

[Fang *et al.*, 2018] Cong Fang, Chris Junchi Li, Zhouchen Lin, and Tong Zhang. Spider: Near-optimal non-convex optimization via stochastic path-integrated differential estimator. *Advances in neural information processing systems*, 31, 2018.

[Feldman and Vondrak, 2019] Vitaly Feldman and Jan Vondrak. High probability generalization bounds for uniformly stable algorithms with nearly optimal rate. In *Conference on Learning Theory*, pages 1270–1279. PMLR, 2019.

[Finn *et al.*, 2017] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pages 1126–1135. PMLR, 2017.

[Foster *et al.*, 2018] Dylan J Foster, Ayush Sekhari, and Karthik Sridharan. Uniform convergence of gradients for non-convex learning and optimization. *Advances in neural information processing systems*, 31, 2018.

[Gao and Huang, 2021] Hongchang Gao and Heng Huang. Fast training method for stochastic compositional optimization problems. *Advances in Neural Information Processing Systems*, 34:25334–25345, 2021.

[Ghadimi and Lan, 2013] Saeed Ghadimi and Guanghui Lan. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM journal on optimization*, 23(4):2341–2368, 2013.

[Ghadimi *et al.*, 2020] Saeed Ghadimi, Andrzej Ruszczynski, and Mengdi Wang. A single timescale stochastic approximation method for nested stochastic optimization. *SIAM Journal on Optimization*, 30(1):960–979, 2020.

[Hardt *et al.*, 2016] Moritz Hardt, Ben Recht, and Yoram Singer. Train faster, generalize better: Stability of stochastic gradient descent. In *International conference on machine learning*, pages 1225–1234. PMLR, 2016.

[Jiang *et al.*, 2022] Wei Jiang, Bokun Wang, Yibo Wang, Lijun Zhang, and Tianbao Yang. Optimal algorithms for stochastic multi-level compositional optimization. In *International Conference on Machine Learning*, pages 10195–10216. PMLR, 2022.

[Johnson and Zhang, 2013] Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. *Advances in neural information processing systems*, 26, 2013.

[Kingma and Ba, 2014] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[Kuzborskij and Lampert, 2018] Ilja Kuzborskij and Christoph Lampert. Data-dependent stability of stochastic gradient descent. In *International Conference on Machine Learning*, pages 2815–2824. PMLR, 2018.

[Lei *et al.*, 2021] Yunwen Lei, Ting Hu, and Ke Tang. Generalization performance of multi-pass stochastic gradient descent with convex loss functions. *Journal of Machine Learning Research*, 22(25):1–41, 2021.

[Lei, 2023] Yunwen Lei. Stability and generalization of stochastic optimization with nonconvex and nonsmooth problems. In *The Thirty Sixth Annual Conference on Learning Theory*, pages 191–227. PMLR, 2023.

[Levy *et al.*, 2021] Kfir Levy, Ali Kavis, and Volkan Cevher. Storm+: Fully adaptive sgd with recursive momentum for nonconvex optimization. *Advances in Neural Information Processing Systems*, 34:20571–20582, 2021.

[Li *et al.*, 2019] Jian Li, Xuanyuan Luo, and Mingda Qiao. On generalization error bounds of noisy gradient methods for non-convex learning. *arXiv preprint arXiv:1902.00621*, 2019.

[Lian *et al.*, 2017] Xiangru Lian, Mengdi Wang, and Ji Liu. Finite-sum composition optimization via variance reduced gradient descent. In *Artificial Intelligence and Statistics*, pages 1159–1167. PMLR, 2017.

[Liu *et al.*, 2024] Jin Liu, Xiaokang Pan, Junwen Duan, Hong-Dong Li, Youqi Li, and Zhe Qu. Faster stochastic variance reduction methods for compositional minimax optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 13927–13935, 2024.

[London *et al.*, 2016] Ben London, Bert Huang, and Lise Getoor. Stability and generalization in structured prediction. *Journal of Machine Learning Research*, 17(221):1–52, 2016.

[Nagarajan and Kolter, 2019] Vaishnavh Nagarajan and J Zico Kolter. Uniform convergence may be unable to explain generalization in deep learning. *Advances in Neural Information Processing Systems*, 32, 2019.

[Nesterov, 1983] Yurii Evgen'evich Nesterov. A method of solving a convex programming problem with convergence rate o\bigl(k^2\bigr). In *Doklady Akademii Nauk*, volume 269, pages 543–547. Russian Academy of Sciences, 1983.

[Nguyen *et al.*, 2017] Lam M Nguyen, Jie Liu, Katya Scheinberg, and Martin Takáč. Sarah: A novel method for machine learning problems using stochastic recursive gradient. In *International conference on machine learning*, pages 2613–2621. PMLR, 2017.

[Qi *et al.*, 2021] Qi Qi, Zhishuai Guo, Yi Xu, Rong Jin, and Tianbao Yang. An online method for a class of distributionally robust optimization with non-convex objectives. *Advances in Neural Information Processing Systems*, 34:10067–10080, 2021.

[Qu *et al.*, 2022] Zhe Qu, Xingyu Li, Rui Duan, Yao Liu, Bo Tang, and Zhuo Lu. Generalized federated learning via sharpness aware minimization. In *International conference on machine learning*, pages 18250–18280. PMLR, 2022.

[Qu *et al.*, 2023] Zhe Qu, Xingyu Li, Xiao Han, Rui Duan, Chengchao Shen, and Lixing Chen. How to prevent the poor performance clients for personalized federated learning? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12167–12176, 2023.

[Rakhlin *et al.*, 2005] Alexander Rakhlin, Sayan Mukherjee, and Tomaso Poggio. Stability results in learning theory. *Analysis and Applications*, 3(04):397–417, 2005.

[Rivasplata *et al.*, 2020] Omar Rivasplata, Ilja Kuzborskij, Csaba Szepesvári, and John Shawe-Taylor. Pac-bayes analysis beyond the usual bounds. *Advances in Neural Information Processing Systems*, 33:16833–16845, 2020.

[Shalev-Shwartz *et al.*, 2010] Shai Shalev-Shwartz, Ohad Shamir, Nathan Srebro, and Karthik Sridharan. Learnability, stability and uniform convergence. *The Journal of Machine Learning Research*, 11:2635–2670, 2010.

[Wang *et al.*, 2017] Mengdi Wang, Ethan X Fang, and Han Liu. Stochastic compositional gradient descent: algorithms for minimizing compositions of expected-value functions. *Mathematical Programming*, 161:419–449, 2017.

[Wang *et al.*, 2024] Shu Wang, Zhe Qu, Yuan Liu, Shichao Kan, Yixiong Liang, and Jianxin Wang. Fedmmr: Multimodal federated learning via missing modality reconstruction. In *2024 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2024.

[Yang *et al.*, 2023] Ming Yang, Xiyuan Wei, Tianbao Yang, and Yiming Ying. Stability and generalization of stochastic compositional gradient descent algorithms. *arXiv preprint arXiv:2307.03357*, 2023.

[Zhang and Xiao, 2019] Junyu Zhang and Lin Xiao. A stochastic composite gradient method with incremental variance reduction. *Advances in Neural Information Processing Systems*, 32, 2019.

[Zhang *et al.*, 2021] Jiayao Zhang, Hua Wang, and Weijie Su. Imitating deep learning dynamics via locally elastic stochastic differential equations. *Advances in Neural Information Processing Systems*, 34:6392–6403, 2021.

[Zhang, 2004] Tong Zhang. Solving large scale linear prediction problems using stochastic gradient descent algorithms. In *Proceedings of the twenty-first international conference on Machine learning*, page 116, 2004.

[Zhou *et al.*, 2018] Dongruo Zhou, Jinghui Chen, Yuan Cao, Yiqi Tang, Ziyan Yang, and Quanquan Gu. On the convergence of adaptive gradient methods for nonconvex optimization. *arXiv preprint arXiv:1808.05671*, 2018.