From Convergence to Generalization: Stability of Stationary-Point Learning Algorithms

Yunwen Lei, *Member, IEEE,* Zimeng Wang and Xiaoming Yuan

***Abstract*—Algorithmic stability is a fundamental concept in learning theory for studying the generalization guarantees of learning algorithms. A notable limitation of classical stability analyses is that they often require convexity assumptions to obtain nontrivial bounds. In this paper, we investigate the stability and generalization properties of learning algorithms in nonconvex settings. We introduce an algorithm-dependent quantity that depends only on the training dataset and the algorithm's output. Under a mild differentiability assumption, we establish stability and generalization bounds that apply to almost any algorithm. Our bounds explicitly involve the optimization error and the algorithm-dependent quantity, thereby capturing the local curvature of the objective function around the learned model. A key feature of our analysis is that it remains valid even when the algorithm does not converge to a global or local minimizer. We further apply our general framework to gradient descent and demonstrate its implications for both linear models and shallow neural networks. Empirical studies verify the effectiveness of our stability analyses.**

***Index Terms*—Learning theory, generalization bound, algorithmic stability, nonconvex learning**

I. Introduction

In machine learning (ML), models are typically trained by applying a learning algorithm to approximately maximize performance on a training dataset. However, good training performance does not necessarily translate into strong performance on unseen test data. In particular, when a model is overly complex, it may fit the training data well while failing to generalize to new samples. The discrepancy between training and testing performance is commonly referred to as the generalization gap. Understanding this generalization gap is a central topic in statistical learning theory (SLT) [1, 2]. A key challenge in this direction is the dependence between the learned model and the training dataset [1, 2], due to which the training error evaluated at the output model is no longer an unbiased estimator of the testing error.

Various techniques have been developed to address this dependence. One of the most widely used approaches is the complexity-based analysis, which controls the uniform deviation between training and testing errors over a hypothesis space in which the learning process operates [3]. This approach typically yields generalization bounds that depend on measures of the hypothesis space complexity, such as the VC dimension, covering numbers, and Rademacher complexity [2, 3]. Another powerful framework for addressing this dependence is algorithmic stability, which quantifies the sensitivity of an algorithm to small perturbations of the training dataset. A foundational result shows that the generalization gap can be controlled by the algorithmic stability [4]. This insight has motivated substantial interest in stability-based analyses of learning algorithms, particularly in stochastic contexts [5]. Beyond these approaches, information-theoretic analyses capture the dependence via mutual information [6, 7], while PAC-Bayesian analyses bound the generalization gap for randomized predictors by controlling the KL divergence between posterior and prior distributions [8, 9].

Early developments in SLT primarily relied on complexity-based approaches, which often fail to explain the remarkable generalization performance of modern ML models. In practice, algorithms are frequently applied to highly overparameterized models, which, according to classical complexity measures, should severely overfit the training data. Contrary to this expectation, empirical evidence shows that such overparameterized models can achieve not only near-zero training error but also strong generalization performance [10]. These observations have spurred growing interest in alternative theoretical frameworks for understanding generalization, including algorithmic stability [5], PAC-Bayesian analysis [8, 9], and information-theoretic approaches [6, 7, 9]. Among these, algorithmic stability offers a particularly appealing perspective, as it explicitly incorporates the optimization trajectory into the generalization analysis. This often leads to tighter and complexity-independent generalization bounds [4, 5]. Such properties make stability-based analyses especially well-suited for overparameterized models, where the number of parameters exceeds the number of training samples.

Algorithmic stability has been widely used to study the generalization behavior of learning algorithms. However, most existing stability analyses rely on convexity assumptions to obtain nontrivial bounds [5, 11, 12]. For example, the stability analysis of stochastic gradient descent (SGD) depends on the nonexpansiveness of the gradient operator [5], which holds only for convex objectives. This significantly limits the applicability of stability-based analysis. Recent work has sought to relax the convexity assumption. One line of research shows that, when the width m of a neural network is sufficiently large, the loss function exhibits a form of weak convexity, with the corresponding weak convexity parameter decaying

Yunwen Lei is with the Department of Mathematics, The University of Hong Kong, Hong Kong, China (e-mail: leiyw@hku.hk). Zimeng Wang is with the Department of Mathematics and Mathematical Statistics, Umeå University, 90187 Umeå, Sweden (e-mail:zimeng.wang@umu.se). Xiaoming Yuan is with the Department of Mathematics, The University of Hong Kong, Hong Kong, China (e-mail: xmyuan@hku.hk). The work of Y. Lei was supported by the Hong Kong Research Grants Council under the GRF projects 17302624 and 17305425. The work of Z. Wang was supported by the Wallenberg AI, Autonomous Systems, and Software Program (WASP), funded by the Knut and Alice Wallenberg Foundation. The work of X. Yuan was supported by the Hong Kong Research Grants Council under the GRF project 17309824.

on the order of $O(1/\sqrt{m})$ [13]. This observation has motivated stability analyses of gradient-based methods for training neural networks [14, 15]. Another line of work imposes the Polyak–Łojasiewicz (PL) condition and demonstrates that any learning algorithm converging to a global minimizer enjoys stability and generalization guarantees [16, 17].

While these relaxations of the convexity assumption significantly broaden the scope of stability analysis, several important limitations remain. For instance, the PL condition is still a strong requirement, as it implies that every stationary point is a global minimizer [18]. Moreover, existing analyses typically require the PL condition to hold uniformly over the entire parameter space [16, 17], which may not accurately reflect the behavior of the algorithm. In many cases, what truly matters is whether a PL-type condition holds along the optimization trajectory rather than globally. Even when the PL condition is satisfied globally, the associated PL constant may be too small to yield tight generalization bounds. Finally, current stability analyses under the PL condition [16, 17] guarantee nontrivial generalization mainly when the algorithm finds a global minimizer, a requirement that is often violated when solving nonconvex problems.

In this paper, we aim to advance the stability analysis of learning algorithms for nonconvex problems. Our main contributions are summarized as follows:

- We introduce an algorithm-dependent quantity as a relaxation of the PL parameter. Unlike the PL parameter, this quantity depends on the output of the algorithm and can be significantly larger, providing a more flexible measure for analyzing stability.
- We show that this algorithm-dependent quantity governs both the stability and generalization of learning algorithms. Our analysis explicitly captures the dependence of generalization on the optimization error and the curvature of the objective around the algorithm's output. Notably, our results require only differentiability and symmetry[1] of the objective and apply to algorithms that output stationary points, in contrast to existing analyses [16, 17], which assume convergence to a global minimizer.
- We further refine the dependency on the optimization error by introducing a one-point quadratic-growth (QG) parameter, which extends and sharpens prior stability analyses under a QG condition [16] by removing several restrictive assumptions and yielding tighter bounds.
- We apply our general results to gradient descent (GD) and derive excess risk bounds of order $O(1/n)$, where n is the sample size. We further demonstrate the practical utility of these bounds by improving existing generalization guarantees for GD applied to linear models and shallow neural networks.
- We conduct empirical analyses to validate our theoretical analyses. Experimental results show our stability bound is indeed an effective control of the generalization gap.

The remainder of the paper is organized as follows. We review related work in Section II and provide background in Section III. Our main results on stability analysis are presented in Section IV, followed by applications in Section V. Experimental results are reported in Section VI. Proofs of the main results are provided in Section VII. We conclude the paper in Section VIII, and defer all remaining experimental analyses and technical proofs to the supplementary material.

[1]The symmetry assumption is not essential and is imposed here to simplify the presentation.

II. Related Work

A. Stability and Generalization

Algorithmic stability quantifies how a small perturbation of the training dataset affects the output of a learning algorithm. One of the most widely studied notions is uniform stability [4], which can be used to derive high-probability generalization bounds [19–21]. Weaker stability notions have also been introduced to establish generalization guarantees under more relaxed conditions. For example, on-average stability studies generalization in expectation [22], and several other notions measure changes in the model parameters themselves, such as argument stability [23] and on-average model stability [12]. For randomized algorithms, stability can be quantified in terms of distances between the probability distributions of output models [24]. Typically, stability-based analyses yield generalization bounds of order $O(1/\sqrt{n})$. Subsequent work has established faster rates of $O(1/n)$ under additional assumptions, such as a low-noise condition [12, 25].

B. Stability Analysis of Gradient-based Methods

An appealing feature of algorithmic stability is that it captures intrinsic properties of the learning algorithm and yields algorithm-dependent generalization bounds. The influential work [5] pioneered the stability analysis of SGD, which has since motivated numerous studies on the stability of other optimization algorithms [12, 16, 26–29]. Optimistic stability bounds have been developed for gradient-based methods applied to both convex models [12] and neural networks [14, 15, 30, 31]. The stability of SGD in nonsmooth settings has also been analyzed [12, 23], while hypothesis stability [16] and on-average stability bounds [17] have been established for black-box optimization methods under the PL condition, which requires the convergence of an algorithm to a global minimizer. Algorithmic stability has been widely employed to study generalization in diverse settings, including meta-learning [32, 33], triplet learning [34], zeroth-order SGD [35], differential privacy [23], delayed SGD [36], biased gradient methods [37], AUPRC maximization [38] and federated learning [39].

III. Background

Let ρ be a probability measure defined on a sample space $\mathcal{Z} := \mathcal{X} \times \mathcal{Y}$, where $\mathcal{X}$ is an input space and $\mathcal{Y} \subset \mathbb{R}$ is an output space. Suppose we draw a training dataset $S = \{\mathbf{z}_1, \ldots, \mathbf{z}_n\}$ independently from ρ, and we want to build a model $h : \mathcal{X} \mapsto \mathcal{Y}$ for prediction. For brevity, we consider parametric models of the form $h_{\mathbf{w}}$, where $\mathbf{w}$ comes from a closed and convex parameter space $\mathcal{W} \subset \mathbb{R}^d$ (d is the dimensionality of the model). We measure the performance of

a model $\mathbf{w}$ on a single example $\mathbf{z}$ via a loss function $f(\mathbf{w};\mathbf{z})$. The performance on the training dataset and on the underlying distribution is quantified by the empirical risk $F_S : \mathcal{W} \to \mathbb{R}$ and the population risk $F : \mathcal{W} \to \mathbb{R}$, respectively:

$$F_S(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^{n} f(\mathbf{w};\mathbf{z}_i), \quad F(\mathbf{w}) = \mathbb{E}_{\mathbf{z}}[f(\mathbf{w};\mathbf{z})],$$

where $\mathbb{E}_{\mathbf{z}}[\cdot]$ denotes the expectation with respect to (w.r.t.) $\mathbf{z}$. Throughout the paper, we assume that $f(\cdot;\mathbf{z})$ is differentiable for every $\mathbf{z} \in \mathcal{Z}$.

We often apply a learning algorithm $A : \cup_{n\in\mathbb{N}}\mathcal{Z}^n \mapsto \mathcal{W}$ to train a model $\mathbf{w} \in \mathcal{W}$ based on a training dataset S. We denote by $A(S)$ the model produced by applying A to S, which may be stochastic (e.g., SGD) or deterministic (e.g., GD). After obtaining $A(S)$, we are interested in its performance relative to the optimal model $\mathbf{w}^* = \arg\min_{\mathbf{w}\in\mathcal{W}} F(\mathbf{w})$. A standard approach is to decompose the excess population risk [40] as

$$\mathbb{E}[F(A(S))] - F(\mathbf{w}^*) = \mathbb{E}[F(A(S)) - F_S(A(S))] + \mathbb{E}[F_S(A(S)) - F_S(\mathbf{w}^*)],$$

where we have used the equality $\mathbb{E}[F_S(\mathbf{w}^*)] = F(\mathbf{w}^*)$ since $\mathbf{w}^*$ is independent of S. We refer to the first term as the *generalization gap*, which measures the difference between the training and testing performance of $A(S)$, and the second term as the *optimization error*, which quantifies the suboptimality of $A(S)$ in terms of the training loss. In this work, we leverage algorithmic stability to control the generalization gap, while the optimization error is analyzed using standard tools from optimization theory.

We say that two datasets S and $\widetilde{S}$ are neighboring datasets if they differ in at most one example. Below, we recall the definition of uniform stability [4].

Definition III.1 (Uniform stability)**.** We say that an algorithm A is ϵ_{unif}-uniformly stable ($\epsilon_{\text{unif}} > 0$) if, for any two neighboring datasets S and $\widetilde{S}$ we have $\sup_{\mathbf{z}} \mathbb{E}_A[|f(A(S);\mathbf{z}) - f(A(\widetilde{S});\mathbf{z})|] \le \epsilon_{\text{unif}}$, where $\mathbb{E}_A[\cdot]$ denotes the expectation taken w.r.t. to the randomness of A.

Uniform stability can be somewhat restrictive since it involves a supremum over both $\mathbf{z}$ and all neighboring datasets. In this paper, we instead consider on-average stability, which measures the average effect of perturbing each training example in the dataset. Let $S = \{\mathbf{z}_1,\dots,\mathbf{z}_n\}$ and $S' = \{\mathbf{z}'_1,\dots,\mathbf{z}'_n\}$ be two datasets of size n drawn independently from ρ. For each $i \in [n] := \{1,\dots,n\}$, define

$$S_i = \{\mathbf{z}_1,\dots,\mathbf{z}_{i-1},\mathbf{z}'_i,\mathbf{z}_{i+1},\dots,\mathbf{z}_n\}. \tag{III.1}$$

Clearly, S_i is a neighboring dataset of S.

Definition III.2 (On-average stability)**.** Let S, S' be two datasets of size n drawn independently from ρ, and let S_i be defined as in Eq. (III.1) for each $i \in [n]$. We say that an algorithm A is on-average ϵ_A-stable ($\epsilon_A > 0$) if

$$\frac{1}{n}\sum_{i=1}^{n} \mathbb{E}_{S,S'}\big[f(A(S_i);\mathbf{z}_i) - f(A(S);\mathbf{z}_i)\big] \le \epsilon_A.$$

The following lemma indicates that on-average stability guarantees generalization in expectation. Note that ϵ_A becomes a random variable if A is a randomized algorithm.

Lemma III.3 ([22])**.** *If an algorithm A is on-average ϵ_A-stable, then $\mathbb{E}_S\big[F(A(S)) - F_S(A(S))\big] \le \epsilon_A$.*

We also recall the definitions of Lipschitz continuity, smoothness, and strong convexity as follows.

Definition III.4 (Smoothness, Lipschitzness, and strong convexity)**.** Let $\mu, L, G \ge 0$. For a differentiable function $g : \mathcal{W} \mapsto \mathbb{R}$, we say g is L-smooth if

$$\|\nabla g(\mathbf{w}) - \nabla g(\mathbf{w}')\|_2 \le L\|\mathbf{w} - \mathbf{w}'\|_2, \ \ \forall \mathbf{w}, \mathbf{w}' \in \mathcal{W}.$$

We say g is G-Lipschitz continuous if $\|\nabla g(\mathbf{w})\|_2 \le G$ for any $\mathbf{w} \in \mathcal{W}$. Moreover, we say g is μ-strongly convex if for any $\mathbf{w}, \mathbf{w}' \in \mathcal{W}$,

$$g(\mathbf{w}) \ge g(\mathbf{w}') + \langle \mathbf{w} - \mathbf{w}', \nabla g(\mathbf{w}')\rangle + \frac{\mu}{2}\|\mathbf{w} - \mathbf{w}'\|_2^2.$$

We say g is convex if the above inequality holds with $\mu = 0$.

IV. Main Results

A. Algorithm-dependent Generalization Bounds

In this subsection, we present a general result characterizing the generalization behavior of learning algorithms that converge to stationary points. First, recall the Polyak-Łojasiewicz (PL) condition as follows.

Definition IV.1 (PL condition)**.** We say that a differentiable function $F_S : \mathcal{W} \to \mathbb{R}$ satisfies the PL condition with parameter $\beta_S > 0$ if

$$F_S(\mathbf{w}) - \hat{F}_S \le \frac{1}{2\beta_S}\|\nabla F_S(\mathbf{w})\|_2^2, \quad \forall \mathbf{w} \in \mathcal{W}, \tag{IV.1}$$

where $\hat{F}_S := \min_{\mathbf{w}\in\mathcal{W}} F_S(\mathbf{w})$.

Remark IV.2. The PL condition is widely used to study the convergence of gradient-based methods [18]. The work [16] identified its role in analyzing the stability of learning algorithms, showing that the hypothesis stability of any algorithm can be controlled by the PL parameter β_S together with the optimization error. This result was further refined in [17], which leveraged the smoothness of the loss functions to derive optimistic on-average stability bounds under the PL condition.

Note that the PL condition is a global property and is thus relatively strong. To increase the applicability of our results, we introduce two weaker notations next. For any S and $\mathbf{w} \in \mathcal{W}$, define $\beta_{S,\mathbf{w}} \in (0, +\infty]$ as

$$\beta_{S,\mathbf{w}} := \frac{\|\nabla F_S(\mathbf{w})\|_2^2}{2\max\{F_S(\mathbf{w}) - F_S(A(S)), 0\}}, \tag{IV.2}$$

where $\|\cdot\|_2$ denotes the Euclidean norm and $\nabla F_S : \mathcal{W} \to \mathbb{R}^d$ is the gradient of F_S. Here we use the maximum operator to ensure that $\beta_{S,\mathbf{w}}$ is nonnegative. Additionally, we define

$$\beta'_{S,\mathbf{w}} := \frac{\|\nabla F_S(\mathbf{w})\|_2^2}{2(F_S(\mathbf{w}) - \hat{F}_S)}. \tag{IV.3}$$

Note $\beta_{S,\mathbf{w}} \geq \beta'_{S,\mathbf{w}}$ for all S and $\mathbf{w}$ due to $\hat{F}_S \leq F_S(A(S))$. Besides, $\beta'_{S,\mathbf{w}}$ is closely related to the PL condition. For a fixed S, if F_S satisfies the PL condition with $\beta_S > 0$, then $\beta'_{S,\mathbf{w}} \geq \beta_S$ for all $\mathbf{w} \in \mathcal{W}$. A notable property is that $\beta'_{S,\mathbf{w}}$ depends on both S and $\mathbf{w}$, the latter of which is often chosen as the output of the algorithm applied to S in the generalization analysis (e.g., we choose $\mathbf{w} = A(S)$). Therefore, $\beta'_{S,\mathbf{w}}$ can be much larger than the PL parameter β_S, and hence so is $\beta_{S,\mathbf{w}}$. In the supplementary material, we provide empirical evidence demonstrating that $\beta_{S,\mathbf{w}}$ is significantly larger than β_S when training shallow neural networks (SNNs).

TABLE I: Summary of notations.

Notation	Meaning	Reference
β_S	PL parameter	Eq. (IV.1)
$\beta_{S,\mathbf{w}}$	$\frac{\|\nabla F_S(\mathbf{w})\|_2^2}{2\max\{F_S(\mathbf{w})-F_S(A(S)),0\}}$	Eq. (IV.2)
$\beta'_{S,\mathbf{w}}$	$\frac{\|\nabla F_S(\mathbf{w})\|_2^2}{2(F_S(\mathbf{w})-\hat{F}_S)}$	Eq. (IV.3)
$\tilde{\beta}_{S,\mathbf{w}}$	a term $\leq \beta'_{S,\mathbf{w}}$	Theorem IV.11
$\tilde{\beta}_S$	$\tilde{\beta}_S \leq \tilde{\beta}_{S,\mathbf{w}_0}/2$	Theorem IV.11
μ_S	$\frac{2(F_S(A(S))-\hat{F}_S)}{\|A(S)-\hat{A}(S)\|_2^2}$	Eq. (IV.10)
μ_{qg}	QG parameter	Eq. (IV.9)

Next, we present our main result in the following theorem. Our analysis clarifies the crucial role of the quantity $\beta_{S,\mathbf{w}}$ on the generalization, which relaxes the PL condition. Some discussions also consider the local PL condition, where Eq. (IV.1) holds around a neighborhood of the initial point $\mathbf{w}_0$ [13]. Geometrically, this implies that any point with a small gradient must lie close (in function value) to the global minimum. In contrast, $\beta_{S,\mathbf{w}}$ compares the gradient to the gap relative to the algorithmic output, i.e., $F_S(\mathbf{w}) - F_S(A(S))$. This leads to two key distinctions. First, if $\beta_{S,\mathbf{w}}$ is bounded from below, points with small gradients can still be far from the global minimum, provided they remain close to the level set of $F_S(A(S))$. Second, while the local PL condition requires Eq. (IV.1) to hold throughout a local neighborhood, our stability bound in Theorem IV.3 only requires $\beta_{S_1,A(S)}$ at the specific point $A(S)$. This makes our approach more flexible for analyzing algorithms that converge to stationary points rather than global minimizers. In the remainder of the paper, when we write $\mathbb{E}[\cdot]$, we mean the expectation taken w.r.t. S, S' and A. We always assume A is symmetric, i.e., the distribution of $A(S)$ does not depend on the order of examples in S. This is a mild assumption that holds for a wide range of algorithms, including GD and SGD.

Theorem IV.3. *Let S_i be defined in Eq.* (III.1) *and A be a symmetric algorithm. Let $\beta_{S,\mathbf{w}}$ be defined in Eq.* (IV.2)*. Then A is on-average ϵ_A-stable with*

$$\epsilon_A = \mathbb{E}\Big[\frac{2\|\nabla f(A(S);\mathbf{z}_1)\|_2^2 + 2\|\nabla f(A(S);\mathbf{z}'_1)\|_2^2}{n\beta_{S_1,A(S)}}\Big] + \mathbb{E}\Big[\frac{n\|\nabla F_S(A(S))\|_2^2}{\beta_{S_1,A(S)}}\Big]. \quad \text{(IV.4)}$$

If F_S is L_S-smooth, then A is on-average ϵ_A-stable with

$$\epsilon_A = \mathbb{E}\Big[\frac{2\|\nabla f(A(S);\mathbf{z}_1)\|_2^2 + 2\|\nabla f(A(S);\mathbf{z}'_1)\|_2^2}{n\beta_{S_1,A(S)}}\Big] + \mathbb{E}\Big[\frac{2L_S n(F_S(A(S)) - \hat{F}_S)}{\beta_{S_1,A(S)}}\Big]. \quad \text{(IV.5)}$$

Furthermore, we have $\mathbb{E}\big[F(A(S)) - F_S(A(S))\big] \leq \epsilon_A$.

Remark IV.4 (Applicability to algorithms finding stationary points). Theorem IV.3 gives stability and generalization bounds for general learning algorithms in terms of $\beta_{S_1,A(S)}$. Eq. (IV.4) is derived without any assumptions except the differentiability of the loss function. It shows that the stability can be bounded by two terms: $\mathbb{E}[(\|\nabla f(A(S);\mathbf{z}_1)\|_2^2 + \|\nabla f(A(S);\mathbf{z}'_1)\|_2^2)/(n\beta_{S_1,A(S)})]$ and $n\mathbb{E}[\|\nabla F_S(A(S))\|_2^2/\beta_{S_1,A(S)}]$. The first term is controlled by the sample size n and the landscape of the problem (reflected by $\beta_{S_1,A(S)}$). On the other hand, the second term is related to the optimization error, which would typically be smaller if we run more iterations. A key difference between our results and the existing stability analysis [16, 17] is that our analysis does not require the algorithm to find a global minimizer. The reason is that Eq. (IV.4) involves the optimization error $\|\nabla F_S(A(S))\|_2^2$, while the generalization bounds in [16, 17] involve $F_S(A(S)) - \hat{F}_S$. Generally, optimization algorithms only guarantee a stationary point for nonconvex problems since the optimization error is measured by the magnitude of the gradients [41]. In this case, Eq. (IV.4) still applies, while the existing stability analysis may not. In particular, if A returns a stationary point, then Eq. (IV.4) implies that

$$\epsilon_A = \mathbb{E}\Big[\frac{2\|\nabla f(A(S);\mathbf{z}_1)\|_2^2 + 2\|\nabla f(A(S);\mathbf{z}'_1)\|_2^2}{n\beta_{S_1,A(S)}}\Big]. \quad \text{(IV.6)}$$

For GD, we can show that $\|\nabla F_S(A(S))\|_2^2$ decays exponentially fast w.r.t. the number of iterations. Then, we can choose a moderate iteration number to make the second term in Eq. (IV.4) dominated by the first term. In this case, we get a generalization bound of the order of $\mathbb{E}[\|\nabla f(A(S);\mathbf{z}'_1)\|_2^2/(n\beta_{S_1,A(S)})]$. In the supplementary material, we empirically verify that $\frac{n\|\nabla F_S(A(S))\|_2^2}{\beta_{S_1,A(S)}}$ remains small when training SNNs with GD.

Remark IV.5 (Comparison with PL condition). Eq. (IV.5) is an extension of the generalization bound developed in [17]. If f is L-smooth with $L \leq n\beta/4$ and

$$\mathbb{E}[F_S(\mathbf{w}) - \hat{F}_S] \leq \frac{1}{2\beta}\mathbb{E}[\|\nabla F_S(\mathbf{w})\|_2^2], \quad \forall \mathbf{w} \in \mathcal{W}, \quad \text{(IV.7)}$$

the following bound was established in [17][2]

$$\mathbb{E}[F(A(S)) - \hat{F}_S] \lesssim \frac{L\mathbb{E}[\hat{F}_S]}{n\beta} + \frac{L\mathbb{E}[F_S(A(S)) - \hat{F}_S]}{\beta}. \quad \text{(IV.8)}$$

Note that Eq. (IV.7) requires a PL parameter β that holds uniformly for all datasets S and all $\mathbf{w} \in \mathcal{W}$. In contrast, Eq. (IV.2) allows the parameter $\beta_{S,A(S)}$ to depend on both

[2]We denote $A \lesssim B$ if there exists a universal constant $c > 0$ such that $A \leq cB$. We denote $A \gtrsim B$ if there exists a universal constant $c' > 0$ such that $A \geq c'B$. We denote $A \asymp B$ if $A \lesssim B$ and $A \gtrsim B$.

the dataset S and the algorithm's output $A(S)$, making it data-dependent. Note $\beta_{S,A(S)}$ is always lower bounded by β and could be significantly larger than β since we have $\beta \leq \min_{\mathbf{w}\in\mathcal{W}} \frac{\mathbb{E}[\|\nabla F_S(\mathbf{w})\|_2^2]}{2(\mathbb{E}[F_S(\mathbf{w})-\hat{F}_S])}$. We shall give explicit estimates of $\beta_{S,A(S)}$ for linear models and SNNs in Section V. Furthermore, the analysis in [17] requires an assumption $L \leq n\beta/4$, which is removed in our analysis. It should be noted that Eq. (IV.8) features a more favorable dependence on the optimization error, as it scales with $L\mathbb{E}[F_S(A(S)) - \hat{F}_S]/\beta$, whereas Eq. (IV.5) scales with $nL\mathbb{E}[(F_S(A(S)) - \hat{F}_S)/\beta_{S_1,A(S)}]$, introducing an extra factor of n. However, this factor does not significantly affect algorithms operating in an interpolation setting. For instance, both SGD and GD are known to achieve linear convergence rates under interpolation [42]. Then, one can simply increase the number of iterations by a logarithmic factor to compensate for the additional n factor.

Remark IV.6. Note that the bound in Eq. (IV.4) is algorithm-dependent since it involves $A(S)$. Furthermore, because $\mathbb{E}[F_S(A(S))] = \mathbb{E}[F_{S_1}(A(S_1))]$, the term $1/\beta_{S_1,A(S)}$ can be approximated as

$$\frac{1}{\beta_{S_1,A(S)}} = \frac{2\max\{F_{S_1}(A(S)) - F_{S_1}(A(S_1)), 0\}}{\|\nabla F_{S_1}(A(S))\|_2^2} \approx \frac{2\max\{F_{S_1}(A(S)) - F_S(A(S)), 0\}}{\|\nabla F_{S_1}(A(S))\|_2^2}.$$

Since $A(S)$ has already been computed by the algorithm, both $F_S(A(S))$ and $F_{S_1}(A(S))$ can be evaluated directly.

Next, we consider the quadratic growth (QG) condition, based on which we introduce a relaxed quantity that leads to improved bounds.

Definition IV.7 (QG condition)**.** We say F_S satisfies the quadratic growth condition with parameter $\mu_{qg} \geq 0$ if

$$F_S(\mathbf{w}) - \hat{F}_S \geq \frac{\mu_{qg}}{2}\|\mathbf{w} - \pi_S(\mathbf{w})\|_2^2, \quad \mathbf{w} \in \mathcal{W}, \tag{IV.9}$$

where $\pi_S : \mathcal{W} \to \mathcal{W}$ denotes the Euclidean projection onto the set of global minimizers of F_S, i.e.,

$$\pi_S(\mathbf{w}) := \arg\min_{\tilde{\mathbf{w}}\in\mathcal{W}} \|\mathbf{w} - \tilde{\mathbf{w}}\|_2, \quad \text{s.t. } F_S(\tilde{\mathbf{w}}) = \hat{F}_S.$$

The QG condition is weaker than the PL condition. Indeed, it has been shown that if a function satisfies the PL condition, then the QG condition holds with the same parameter [18]. Similar to the PL condition, the QG condition is also a global property. To relax it, we introduce the one-point QG parameter $\mu_S \geq 0$ defined by

$$\mu_S := \frac{2(F_S(A(S)) - \hat{F}_S)}{\|A(S) - \widetilde{A}(S)\|_2^2}, \tag{IV.10}$$

where the algorithm $\widetilde{A}$ projects the output of A onto the set of global minimizers, i.e.,

$$\widetilde{A}(S) := \pi_S(A(S)). \tag{IV.11}$$

Note that $\widetilde{A}$ is symmetric if A is symmetric. Eq. (IV.10) gives a localization of Eq. (IV.9) at $\mathbf{w} = A(S)$ and hence $\mu_S \geq \mu_{qg}$. In the supplementary material, we provide empirical verification to show that μ_S can be significantly larger than μ_{qg}.

In Theorem IV.8, we leverage μ_S to derive a population risk bound with improved dependency on the optimization error.

Theorem IV.8. *Let S_i be defined in Eq.* (III.1) *and A be a symmetric algorithm. Let $\beta_{S,\mathbf{w}}$ be defined in Eq.* (IV.2)*. If the map $\mathbf{w} \mapsto f(\mathbf{w};\mathbf{z})$ is L-smooth for any $\mathbf{z}$, then*

$$\begin{aligned}&\mathbb{E}[F(A(S)) - \hat{F}_S] \\ &\lesssim \mathbb{E}\Big[\frac{\|\nabla f(A(S);\mathbf{z}_1)\|_2^2 + \|\nabla f(A(S);\mathbf{z}_1')\|_2^2}{n\beta_{S_1,\widetilde{A}(S)}}\Big] \\ &\quad + L\mathbb{E}\Big[\|A(S) - \widetilde{A}(S)\|_2^2\Big(1 + \frac{L}{n\beta_{S_1,\widetilde{A}(S)}}\Big)\Big].\end{aligned} \tag{IV.12}$$

Moreover, consider μ_S defined in Eq. (IV.10)*, then*

$$\begin{aligned}&\mathbb{E}[F(A(S)) - \hat{F}_S] \\ &\lesssim \mathbb{E}\Big[\frac{\|\nabla f(A(S);\mathbf{z}_1)\|_2^2 + \|\nabla f(A(S);\mathbf{z}_1')\|_2^2}{n\beta_{S_1,\widetilde{A}(S)}}\Big] \\ &\quad + L\mathbb{E}\Big[\Big(\frac{F_S(A(S)) - \hat{F}_S}{\mu_S}\Big)\Big(1 + \frac{L}{n\beta_{S_1,\widetilde{A}(S)}}\Big)\Big].\end{aligned} \tag{IV.13}$$

Remark IV.9 (Comparison with QG condition). Theorem IV.8 improves upon Theorem IV.3 by replacing $n(F_S(A(S)) - \hat{F}_S)$ with $F_S(A(S)) - \hat{F}_S$. Therefore, it requires less accuracy in optimization to achieve similar risk bounds. If we further assume $L \lesssim n\beta_{S_1,\widetilde{A}(S)}$, then Eq. (IV.13) implies

$$\begin{aligned}\mathbb{E}[F(A(S)) - \hat{F}_S] &\lesssim L\mathbb{E}\Big[\frac{F_S(A(S)) - \hat{F}_S}{\mu_S}\Big] \\ &\quad + \mathbb{E}\Big[\frac{\|\nabla f(A(S);\mathbf{z}_1)\|_2^2 + \|\nabla f(A(S);\mathbf{z}_1')\|_2^2}{n\beta_{S_1,\widetilde{A}(S)}}\Big].\end{aligned} \tag{IV.14}$$

Under a QG condition and a G-Lipschitzness condition, together with the assumptions that $f(\mathbf{w};\mathbf{z}) \leq c$ for all $\mathbf{w},\mathbf{z}$ and $\pi_S(A(S)) = \pi_S(A(\widetilde{S}))$ for neighboring datasets $S, \widetilde{S}$, the following uniform stability bound was developed in [16]

$$\epsilon_{\text{unif}} \lesssim G\Big(\frac{F_S(A(S)) - \hat{F}_S}{\mu_{qg}}\Big)^{\frac{1}{2}} + G\Big(\frac{c}{\mu_{qg}n}\Big)^{\frac{1}{2}}. \tag{IV.15}$$

We compare this bound with Eq. (IV.14) from three aspects. First, Eq. (IV.15) only guarantees a bound of order $1/\sqrt{n}$, while Eq. (IV.14) shows a faster bound of order $O(1/n)$. Second, Eq. (IV.15) requires the QG parameter μ_{qg} to hold for any $\mathbf{w} \in \mathcal{W}$. As a comparison, Eq. (IV.14) only involves μ_S depending on $A(S)$, which could be much larger than μ_{qg} in practice. Third, Eq. (IV.15) requires additional assumptions such as the boundedness of f and $\pi_S(A(S)) = \pi_S(A(\widetilde{S}))$ for any neighboring $S, \widetilde{S}$. In particular, the latter assumption almost means that there is a unique global minimizer, which is restrictive in nonconvex settings. We remove these assumptions entirely, and obtain bounds involving potentially large parameters $\beta_{S_1,\widetilde{A}(S)}$ and μ_S.

B. Generalization Bounds of Gradient Descent

We proceed to apply the general results obtained in Section IV-A to study the generalization performance of models trained by gradient descent (GD). We assume $\mathcal{W} = \mathbb{R}^d$ hereafter.

Definition IV.10 (Gradient Descent)**.** Let $\mathbf{w}_0 \in \mathbb{R}^d$ and $\eta > 0$ be the step size. Given a dataset S, GD updates the model via

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \nabla F_S(\mathbf{w}_t), \quad t \geq 0. \tag{GD}$$

To conduct analysis, we introduce a new quantity $\tilde{\beta}_{S,\mathbf{w}}$, which is no larger than $\beta'_{S,\mathbf{w}}$ given by Eq. (IV.3) for all S and $\mathbf{w} \in \mathbb{R}^d$. We impose an assumption that the parameter $\tilde{\beta}_{S,\mathbf{w}}$ is Lipschitz continuous w.r.t. $\mathbf{w}$ and a constraint on $\tilde{\beta}_{S,\mathbf{w}_0}$ that aims to make $\beta'_{S,\mathbf{w}}$ larger than $\tilde{\beta}_S$ in a neighborhood around $\mathbf{w}_0$ for some $\tilde{\beta}_S > 0$. In Section V, we shall verify these requirements with appropriately defined $\tilde{\beta}_{S,\mathbf{w}}$ for both linear models and SNNs. Theorem IV.11 gives population risk bounds for (GD) after a logarithmic number of iterations $T > 0$. Note that T benefits from large $\tilde{\beta}_S$. This parameter is related to $\tilde{\beta}_{S,\mathbf{w}_0}$, which captures the curvature at $\mathbf{w}_0$ and could be much larger than the PL parameter in Definition IV.1.

Theorem IV.11. *Let $\mathbf{w}_0 \in \mathbb{R}^d$ be an initial model. For any $S, \mathbf{w}$, let $\beta'_{S,\mathbf{w}}$ be defined by Eq.* (IV.3) *and $\tilde{\beta}_{S,\mathbf{w}} \leq \beta'_{S,\mathbf{w}}$ for some $\tilde{\beta}_{S,\mathbf{w}} > 0$. Assume there exists $L_\beta > 0$ such that*

$$|\tilde{\beta}_{S,\mathbf{w}} - \tilde{\beta}_{S,\mathbf{w}'}| \leq L_\beta \|\mathbf{w} - \mathbf{w}'\|_2. \tag{IV.16}$$

Let $\tilde{\beta}_S > 0$ be chosen such that $\tilde{\beta}_S \leq \tilde{\beta}_{S,\mathbf{w}_0}/2$. Suppose $L_\beta \sqrt{8(F_S(\mathbf{w}_0) - \hat{F}_S)/\tilde{\beta}_S} \leq \tilde{\beta}_S$ and F_S is L_S-smooth. Let $\{\mathbf{w}_t\}_{t\in\mathbb{N}}$ be produced by (GD) *with $\eta = 1/L_S$ and $A(S) = \mathbf{w}_T$. If $T \geq \frac{L_S}{\tilde{\beta}_S} \log(n^2 L_S(F_S(\mathbf{w}_0) - \hat{F}_S))$, then*

$$\mathbb{E}[F(A(S)) - \hat{F}_S] \lesssim \frac{1}{n^2 L_S} + \frac{1}{n}\mathbb{E}\Big[\frac{\|\nabla f(A(S);\mathbf{z}_1)\|_2^2 + \|\nabla f(A(S);\mathbf{z}'_1)\|_2^2 + 1}{\beta_{S_1,A(S)}}\Big]. \tag{IV.17}$$

The next theorem derives similar risk bounds for (GD) with fewer iterations by using the one-point QG parameter μ_S defined in Eq. (IV.10). Additionally, the risk bounds involve $\beta_{S_1,\widetilde{A}(S)}$ rather than $\beta_{S_1,A(S)}$.

Theorem IV.12. *Consider the same setting as in Theorem IV.11. Let μ_S be defined in Eq.* (IV.10)*. Assume that the map $\mathbf{w} \mapsto f(\mathbf{w};\mathbf{z})$ is L-smooth for any $\mathbf{z} \in \mathcal{Z}$ and $L \lesssim n\beta_{S_1,\widetilde{A}(S)}$. If*

$$T \geq \frac{L_S}{\tilde{\beta}_S} \log\big(nL\beta_{S_1,\widetilde{A}(S)}(F_S(\mathbf{w}_0) - \hat{F}_S)/\mu_S\big), \tag{IV.18}$$

then

$$\mathbb{E}[F(A(S)) - \hat{F}_S] \lesssim \frac{1}{n}\mathbb{E}\Big[\frac{\|\nabla f(A(S);\mathbf{z}_1)\|_2^2 + \|\nabla f(A(S);\mathbf{z}'_1)\|_2^2 + 1}{\beta_{S_1,\widetilde{A}(S)}}\Big]. \tag{IV.19}$$

Remark IV.13. Note that the assumption $L \lesssim n\beta_{S_1,\widetilde{A}(S)}$ is neither restrictive (since n can be very large) nor essential: even without it, a similar result can be obtained by invoking Eq. (IV.13) instead of Eq. (IV.14). Moreover, although we focus on GD in this section, our analysis applies to any symmetric optimization algorithm. One can directly substitute the corresponding optimization error bounds into Theorem IV.3 or Theorem IV.8 to derive the associated population risk bounds. We omit the details of these extensions for brevity.

V. Applications

We now apply the general results obtained for GD to linear models and SNNs. For simplicity, we assume $L \lesssim n\beta_{S_1,\widetilde{A}(S)}$. We will demonstrate empirically in the supplementary material that this is a mild assumption.

A. Linear Models

We first consider linear models $\mathbf{x} \mapsto \mathbf{w}^\top \mathbf{x}$, for which the loss function takes the form $f(\mathbf{w};\mathbf{z}) := \ell_y(\mathbf{w}^\top \mathbf{x})$ for $\mathbf{z} := (\mathbf{x}, y)$. Assume that ℓ_y is μ_ℓ-strongly convex and L_ℓ-smooth for any $y \in \mathbb{R}$, then it can be shown that [17, 18]

$$\beta_{S,\mathbf{w}} \geq \beta'_{S,\mathbf{w}} \geq \mu_\ell \sigma_{\min}(\Sigma_S) \quad \forall \mathbf{w} \in \mathcal{W}, \tag{V.1}$$

where $\sigma_{\min}(\cdot)$ denotes the minimal singular value of a matrix and $\Sigma_S = \frac{1}{n}\sum_{i=1}^n \mathbf{x}_i \mathbf{x}_i^\top \in \mathbb{R}^{d\times d}$. Therefore, we can choose $\tilde{\beta}_{S,\mathbf{w}} = \mu_\ell \sigma_{\min}(\Sigma_S)$, which satisfies Eq. (IV.16) with $L_\beta = 0$. We can also choose $\tilde{\beta}_S = \mu_\ell \sigma_{\min}(\Sigma_S)/2$. Assume $\|\mathbf{x}\|_2 \leq 1$ for any $\mathbf{x} \in \mathcal{X}$, then it is clear that $f(\mathbf{w};\mathbf{z}) = \ell_y(\mathbf{w}^\top \mathbf{x})$ is L_ℓ-smooth. Recall that L_S is the smoothness parameter of F_S. The following result follows directly from Theorem IV.12 and Eq. (V.1). We omit its proof for brevity.

Theorem V.1. *Consider the loss function $f(\mathbf{w};\mathbf{z}) = \ell_y(\mathbf{w}^\top \mathbf{x})$. Let $\tilde{\beta}_S = \mu_\ell \sigma_{\min}(\Sigma_S)/2$ and $L = L_\ell$. Let $\{\mathbf{w}_t\}_{t\in\mathbb{N}}$ be produced by* (GD) *with $\eta = 1/L_S$ and $A(S) = \mathbf{w}_T$. If T satisfies Eq.* (IV.18)*, then we have*

$$\mathbb{E}[F(A(S)) - \hat{F}_S] \lesssim \frac{1}{n}\mathbb{E}\Big[\frac{\|\nabla f(A(S);\mathbf{z}_1)\|_2^2 + \|\nabla f(A(S);\mathbf{z}'_1)\|_2^2 + 1}{\mu_\ell \sigma_{\min}(\Sigma_{S_1})}\Big].$$

Remark V.2. Similar generalization bounds were derived for the empirical risk minimizer (ERM) of linear models [43]. Indeed, if ℓ_y is ρ-Lipschitz and μ_ℓ-strongly convex, then for A being the ERM, it was shown that [43]

$$\mathbb{E}[F(A(S)) - \hat{F}_S] \leq \mathbb{E}\Big[\frac{2\rho^2 \mathrm{tr}(\Sigma_S)}{n\mu_\ell \sigma_{\min}(\Sigma_S)}\Big],$$

where $\mathrm{tr}(\cdot)$ denotes the trace operator of a square matrix. Theorem V.1 extends the result to GD for linear models, while removing the Lipschitzness and ERM assumptions.

B. Shallow Neural Networks

Consider SNNs with a single hidden layer with d inputs, m hidden neurons, and a single output neuron. Let $\phi : \mathbb{R} \mapsto \mathbb{R}$ be an activation. The prediction function is of the form

$$h_{\mathbf{v},\mathbf{w}}(\mathbf{x}) = \sum_{k=1}^m v_k \phi\big(\langle \mathbf{w}^{(k)}, \mathbf{x}\rangle\big) := \mathbf{v}^\top \Phi(\mathbf{w}\mathbf{x}), \tag{V.2}$$

where $\Phi : \mathbb{R}^m \to \mathbb{R}^m$ applies ϕ elementwise, $\mathbf{w} := (\mathbf{w}^{(1)}, \ldots, \mathbf{w}^{(m)})^\top \in \mathbb{R}^{m\times d}$ and $\mathbf{v} := (v_1, \ldots, v_m)^\top \in \mathbb{R}^m$ are the weights of the hidden and output neurons, respectively. Analogous to the work [44, 45], we fix $\mathbf{v}$ as a constant vector with $|v_k| = 1/\sqrt{m}$ and train $\mathbf{w}$ with the mean squared error (MSE) loss $f(\mathbf{w};\mathbf{z}) = \frac{1}{2}\big(\mathbf{v}^\top \Phi(\mathbf{w}\mathbf{x}) - y\big)^2$ on S. We always

assume that the feature vectors $\{\mathbf{x}_i\}_{i=1}^n$ have unit ℓ_2-norm and collect the input data into a data matrix

$$X = \big(\mathbf{x}_1, \ldots, \mathbf{x}_n\big)^\top \in \mathbb{R}^{n\times d}.$$

For brevity, we always assume $\hat{F}_S = 0$ in this subsection, i.e., there is an SNN with perfect fitting to training examples.

To find a Lipschitz $\tilde{\beta}_{S,\mathbf{w}}$, we impose Assumption V.3 on ϕ, which is widely used in the theoretical analysis of neural networks [45]. Activation functions satisfying Assumption V.3 include the sigmoid and hyperbolic tangent activations.

Assumption V.3. There exist constants $B_1, B_2 > 0$ such that ϕ is B_1-Lipschitz continuous and B_2-smooth.

As shown in [46], SNNs defined previously satisfy

$$F_S(\mathbf{w}) \le \|\nabla F_S(\mathbf{w})\|_F^2/(2\tilde{\beta}_{S,\mathbf{w}}).$$

Here, $\|\cdot\|_F$ denotes the Frobenius norm of a matrix and

$$\tilde{\beta}_{S,\mathbf{w}} := \frac{1}{n}\sigma_{\min}\Big(\frac{1}{m}\sum_{k=1}^m \Big(\Phi'(X\mathbf{w}^{(k)})\big(\Phi'(X\mathbf{w}^{(k)})\big)^\top\Big)\odot(XX^\top)\Big), \tag{V.3}$$

where Φ' applies ϕ' elementwise and $\odot$ is the Hadamard (elementwise) product of two matrices. Hence it follows that $\beta_{S,\mathbf{w}} \ge \beta'_{S,\mathbf{w}} \ge \tilde{\beta}_{S,\mathbf{w}}$. Furthermore, we know f is L-smooth with L independent of n and m [14]. The following lemma shows that $\tilde{\beta}_{S,\mathbf{w}}$ is Lipschitz continuous w.r.t. $\mathbf{w}$.

Lemma V.4. *Let $\tilde{\beta}_{S,\mathbf{w}}$ be defined in Eq.* (V.3)*. If Assumption V.3 holds, then we have*

$$\big|\tilde{\beta}_{S,\mathbf{w}} - \tilde{\beta}_{S,\tilde{\mathbf{w}}}\big| \le \frac{2B_1B_2\|X\|^2}{n\sqrt{m}}\big\|\mathbf{w}-\tilde{\mathbf{w}}\big\|_F.$$

The following theorem gives high probability population risk bounds for SNNs trained by GD. The condition on the number of hidden neurons m is imposed to ensure that the iterates of GD fall inside a local ball around the initial point $\mathbf{w}_0 \in \mathbb{R}^{m\times d}$, which is required to apply a result in [42] that gives linear convergence of the optimization errors for GD. Note this condition is not needed for the stability analysis.

Theorem V.5. *Let Assumption V.3 hold. Let $\{\mathbf{w}_t\}_{t\in\mathbb{N}}$ be produced by* (GD) *with $\eta = 1/L_S$ and $A(S) = \mathbf{w}_T$. Suppose $m \ge 32B_1^2B_2^2\|X\|^4F_S(\mathbf{w}_0)/(n^2\tilde{\beta}_S^3)$ and T satisfies Eq.* (IV.18)*. Define*

$$\tilde{\beta}_S := \frac{1}{4n}\sigma_{\min}\Big(\mathbb{E}_{\mathbf{w}^{(1)}}\big[\Phi'(X\mathbf{w}^{(1)})\big(\Phi'(X\mathbf{w}^{(1)})\big)^\top\big]\odot(XX^\top)\Big),$$

where we assume $\mathbf{w}_0^{(k)} \overset{i.i.d.}{\sim} \mathbf{w}^{(1)}$ for each $k \in [m]$. Then with probability (w.r.t. the initialization of $\mathbf{w}_0$) at least $1 - n\exp\big(-(2B_1^2\|X\|^2)^{-1}mn\tilde{\beta}_S\big)$, we have

$$\mathbb{E}[F(A(S))] \lesssim \mathbb{E}\Big[\frac{\|\nabla f(A(S);\mathbf{z}_1)\|_2^2 + \|\nabla f(A(S);\mathbf{z}_1')\|_2^2 + 1}{n\beta_{S_1,\widetilde{A}(S)}}\Big].$$

Remark V.6. If $n \gtrsim d$, then we expect that $\|X\|$ grows at a rate of $O(\sqrt{n/d})$ [45]. Then the overparameterization condition in Theorem V.5 becomes $m \gtrsim 1/(d^2\tilde{\beta}_S^3)$. If $n \lesssim d$, then $\|X\|$ is of order of 1 and the overparameterization condition becomes $m \gtrsim 1/(n^2\tilde{\beta}_S^3)$.

Remark V.7. Linear convergence of GD has been developed for training SNNs [13, 45], which focus on optimization. Theorem V.5 extends these convergence rates to risk bounds, which reflect the behavior of models on testing examples. Stability and generalization of GD were also studied recently for SNNs [14, 15, 47, 48]. Assuming for some $\alpha \in [0,1]$ that

$$\min_{\mathbf{w}} F(\mathbf{w}) + \lambda\|\mathbf{w}-\mathbf{w}_0\|_2^2 \lesssim \lambda^\alpha \tag{V.4}$$

and $F(\mathbf{w}^*) = 0$, it has been shown that $\mathbb{E}[F(\mathbf{w}_T)] \lesssim n^{-\alpha}$ if $T \asymp n$ and $m \asymp n^3$ [47]. As a comparison, our analysis does not require the assumption in Eq. (V.4) and imposes a milder overparameterization condition on m (see Remark V.6). Furthermore, the analysis in [47] derives a sublinear convergence and requires $T \asymp n$ to get good generalization. As a comparison, our analysis uses the linear convergence of GD, which shows that $T \gtrsim 1/\tilde{\beta}_S$ (up to log factors) is enough to get models with good prediction. The analysis [15, 30, 48] requires the loss function to satisfy a self-bounding weak-convexity assumption, which does not hold for the least square loss considered here. Furthermore, their analysis imposes an assumption that the dataset is separable by a neural tangent kernel model with a positive margin, which is not required here. While the analysis in [17] also implies risk bounds of order $O(1/n)$, they require Eq. (IV.7) to hold for a β applicable to all $\mathbf{w}$. As a comparison, the risk bounds in Theorem V.5 depend on a much larger quantity $\beta_{S_1,\widetilde{A}(S)}$, and therefore could be significantly better.

VI. Experiments

We nowconduct empirical studies to validate the on-average stability analysis established in our main result, Theorem IV.3. Specifically, we consider shallow neural networks (SNNs) with a single hidden layer containing m neurons. The network is trained using the mean squared error (MSE) loss

$$f(\mathbf{w};\mathbf{z}) = \frac{1}{2}(h_{\mathbf{w}}(\mathbf{x}) - y)^2. \tag{VI.1}$$

Here, $h_{\mathbf{w}}(\cdot)$ denotes the output of the SNN defined in Eq. (V.2) with a fixed $\mathbf{v} = (v_1, \ldots, v_m)^\top \in \mathbb{R}^m$, where each v_k is independently sampled from $\{1/\sqrt{m}, -1/\sqrt{m}\}$ with equal probability. To satisfy Assumption V.3, we choose the activation function ϕ to be the sigmoid function. Moreover, $\mathbf{z} := (\mathbf{x}, y) \in \mathbb{R}^d \times \mathbb{R}$ denotes a data sample, where $\mathbf{x} \in \mathbb{R}^d$ is the feature vector and $y \in \mathbb{R}$ is the target variable. Since Theorem IV.3 applies to all symmetric learning algorithms, we train all models using mini-batch SGD. In particular, mini-batch SGD randomly samples indices from the *uniform distribution* over $[n]$ at each iteration. As a result, the ordering of examples in the dataset S does not affect the distribution of the output $A(S)$, since uniform sampling treats all indices identically.

To conduct the experiments, we use four real-world datasets from the LIBSVM library [49], whose information is summarized in Table II. All datasets are preprocessed so that the feature vectors $\mathbf{x}$ are normalized to have unit ℓ_2-norm. The target variables are rescaled to $y \in [0,1]$ for regression datasets and $y \in \{0,1\}$ for binary classification datasets. For each dataset, we randomly split the samples into a training set S

Dataset	*cadata*	*cpusmall*	*ijcnn*1	*covtype*
n	20640	8192	49990	581012
d	8	12	22	54
Type	Regression	Regression	Classification	Classification

TABLE II: Summary of datasets used in the experiments. Here, 'n' means sample size and 'd' means number of features.

containing 80% of the data and a testing set S_{test} containing the remaining 20%.

Our experiments consist of two main parts: (i) we compare the on-average stability ϵ_A defined in Eq. (IV.4) with the generalization error estimated from the training and testing datasets; and (ii) we investigate the relationship between the on-average stability ϵ_A at stationary points and sample size n.

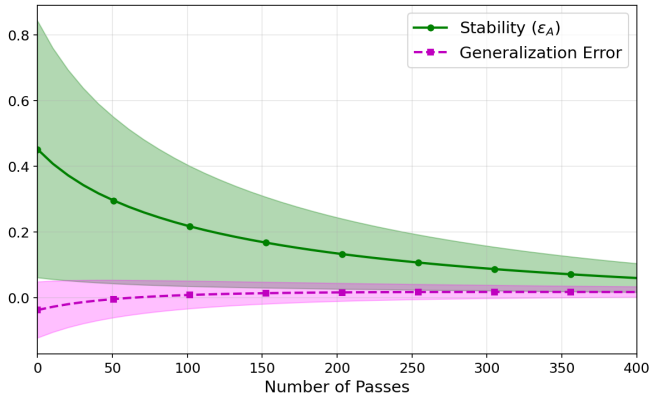
A. Comparison between Stability and Generalization Error

We first compare the on-average stability ϵ_A defined in Eq. (IV.4) with the generalization error. As indicated by Lemma III.3, ϵ_A serves as an upper bound on the generalization error. To verify this empirically, we train an SNN with $m = 1000$ hidden neurons on the *cpusmall* and *cadata* datasets using mini-batch SGD. For both datasets, we record the iterates generated with a constant learning rate $\eta = 0.1$ and a batch size of 128. To ensure robustness and statistical reliability, we conduct 40 independent repetitions for each test, with the initial point $\mathbf{w}_0$ in each repetition drawn from a standard Gaussian distribution. To approximately compute ϵ_A in Eq. (IV.4), we randomly select one sample from the test set S_{test} as $\mathbf{z}_1'$ and construct a neighboring dataset S_1 by replacing the first sample in S with $\mathbf{z}_1'$. The generalization error is approximated by $F_{S_{test}}(\mathbf{w}_t) - F_S(\mathbf{w}_t)$, which is an unbiased estimator of the true generalization error.

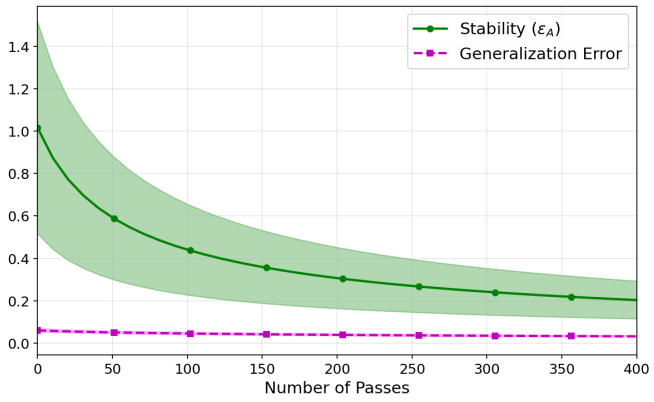
In Figure 1, we report the evolution of the mean and standard deviation of the stability ϵ_A and the generalization error as functions of the number of passes (defined as $t \times \text{batchsize}/n$) during training for both datasets. The results show that the stability ϵ_A given in Eq. (IV.4) consistently upper bounds the generalization error throughout the training process. Moreover, ϵ_A becomes increasingly close to the generalization error as training progresses. These observations validate Theorem IV.3 and indicate that the bound ϵ_A is relatively tight. In addition, we observe that ϵ_A decreases over the course of training. This behavior can be explained by the fact that SGD tends to converge to a stationary point of the empirical loss F_S, for which $\|\nabla F_S(\mathbf{w}_t)\|_2^2$ becomes small as t grows. Consequently, the second term in Eq. (IV.4) decreases along the training trajectory.

B. Test on Training Sample Sizes

In this part, we investigate how the stability ϵ_A defined in Eq. (IV.4) scales with the number of training samples. As discussed in Remark IV.4, when the optimization algorithm reaches a stationary point, the second term in Eq. (IV.4) vanishes, leading to the simplified stability bound in Eq. (IV.6), which improves as n increases. We verify this prediction



(a) *cpusmall* dataset



(b) *cadata* dataset

Fig. 1: Evolution of on-average stability ϵ_A and generalization error during training. Solid and dashed lines record mean values over 40 random repetitions, with shaded regions indicating ± 1 standard deviation.

through experiments with varying training set sizes n. Specifically, we use mini-batch SGD to train an SNN model on the datasets *ijcnn*1 and *covtype*. For both datasets, the learning rate is set to $\eta = 0.02$ and the batch size is fixed at 32. We consider logarithmically spaced sample sizes n ranging from 1000 to 40000. For each value of n, we run mini-batch SGD until obtaining an iterate $\mathbf{w}_T$ that satisfies the stationarity condition $\|\nabla F_S(\mathbf{w}_T)\|_2 \leq 10^{-6}$. All other experimental settings are the same as in the previous experiment.

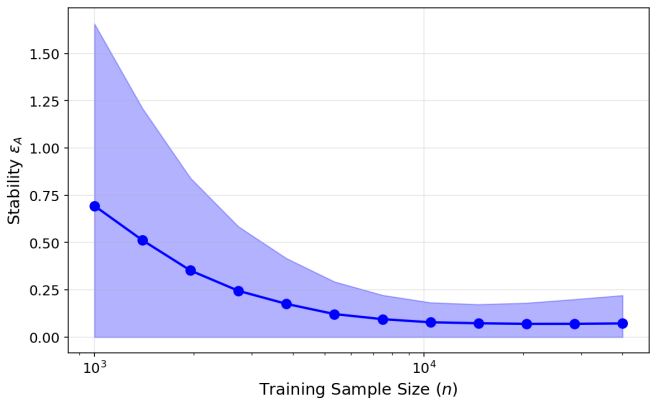
Figure 2 shows how the training sample size n influences the stability ϵ_A at the (nearly) stationary point $\mathbf{w}_T$. We observe that ϵ_A decreases as n increases for both datasets, which is consistent with our prediction that stability improves with larger training sets.

VII. Proofs on Main Generalization Results

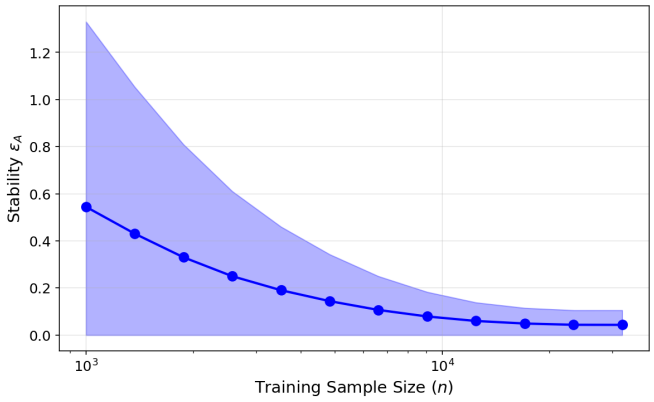
In this section, we present the proofs of Theorems IV.3 and IV.8. We start by introducing the following self-bounding property of smooth functions, which shows that the gradient norms can be bounded by the function values.

Lemma VII.1 ([50]). *If $g : \mathbb{R}^d \to \mathbb{R}$ is L-smooth, then*

$$\|\nabla g(\mathbf{w})\|_2^2 \leq 2L\big(g(\mathbf{w}) - \min_{\mathbf{w}' \in \mathbb{R}^d} g(\mathbf{w}')\big), \quad \forall \mathbf{w} \in \mathbb{R}^d.$$



(a) $ijcnn1$ dataset



(b) *covtype* dataset

Fig. 2: On-average stability ϵ_A at stationary points as a function of training sample size n. Solid lines show mean values over 40 random repetitions, with shaded regions indicating ± 1 standard deviation.

Proof of Theorem IV.3. In the proof, we denote by $\mathbb{E}[\cdot]$ the expectation w.r.t. both the datasets S, S' and the learning algorithm A. Given a training set S, it follows from the definitions of F_S and S_i (Eq. (III.1)) that

$$f(A(S_i);\mathbf{z}_i) = nF_S(A(S_i)) - nF_{S_i}(A(S_i)) + f(A(S_i);\mathbf{z}_i'). \tag{VII.1}$$

Since $\mathbf{z}_i$ and $\mathbf{z}_i'$ follow from the same distribution, we have $\mathbb{E}[f(A(S_i);\mathbf{z}_i')] = \mathbb{E}[f(A(S);\mathbf{z}_i)]$. Hence, taking expectations on both sides of Eq. (VII.1), we obtain that

$$\begin{aligned}\mathbb{E}\big[f(A(S_i);\mathbf{z}_i)\big] &= n\mathbb{E}\big[F_S(A(S_i))\big] \\ &\quad - n\mathbb{E}\big[F_{S_i}(A(S_i))\big] + \mathbb{E}\big[f(A(S);\mathbf{z}_i)\big],\end{aligned}$$

which together with $\mathbb{E}[F_{S_i}(A(S_i))] = \mathbb{E}[F_S(A(S))]$ due to the symmetry between $\mathbf{z}_i$ and $\mathbf{z}_i'$ further implies

$$\begin{aligned}&\mathbb{E}\big[f(A(S_i);\mathbf{z}_i) - f(A(S);\mathbf{z}_i)\big] = n\mathbb{E}\big[F_S(A(S_i)) - F_{S_i}(A(S_i))\big] \\ &= n\mathbb{E}\big[F_S(A(S_i)) - F_S(A(S))\big].\end{aligned} \tag{VII.2}$$

On the other hand, we know from the definition of $\beta_{S,\mathbf{w}}$ in Eq. (IV.2) that

$$\begin{aligned}\frac{1}{\beta_{S,A(S_i)}} &= \frac{2\max\{F_S(A(S_i)) - F_S(A(S)), 0\}}{\|\nabla F_S(A(S_i))\|_2^2} \\ &\geq \frac{2(F_S(A(S_i)) - F_S(A(S)))}{\|\nabla F_S(A(S_i))\|_2^2}.\end{aligned} \tag{VII.3}$$

Rearranging the terms in the above inequality gives

$$F_S(A(S_i)) - F_S(A(S)) \leq \frac{\|\nabla F_S(A(S_i))\|_2^2}{2\beta_{S,A(S_i)}},$$

applying which in Eq. (VII.2) further implies that

$$\begin{aligned}&\mathbb{E}\big[f(A(S_i);\mathbf{z}_i) - f(A(S);\mathbf{z}_i)\big] \\ &\qquad \leq \frac{n}{2}\mathbb{E}\big[\|\nabla F_S(A(S_i))\|_2^2/\beta_{S,A(S_i)}\big].\end{aligned} \tag{VII.4}$$

To estimate the RHS of Eq. (VII.4), we apply the basic inequality $(a+b)^2 \leq 2(a^2+b^2)$ twice to obtain

$$\begin{aligned}&\|\nabla F_S(A(S_i))\|_2^2 \\ &= \big\|\nabla F_S(A(S_i)) - \nabla F_{S_i}(A(S_i)) + \nabla F_{S_i}(A(S_i))\big\|_2^2 \\ &\leq 2\Big\|\nabla F_S(A(S_i)) - \nabla F_{S_i}(A(S_i))\Big\|_2^2 + 2\|\nabla F_{S_i}(A(S_i))\|_2^2 \\ &= 2\Big\|\frac{1}{n}\nabla f(A(S_i);\mathbf{z}_i) - \frac{1}{n}\nabla f(A(S_i);\mathbf{z}_i')\Big\|_2^2 + 2\|\nabla F_{S_i}(A(S_i))\|_2^2 \\ &\leq \frac{4\|\nabla f(A(S_i);\mathbf{z}_i')\|_2^2}{n^2} + \frac{4\|\nabla f(A(S_i);\mathbf{z}_i)\|_2^2}{n^2} + 2\|\nabla F_{S_i}(A(S_i))\|_2^2.\end{aligned} \tag{VII.5}$$

Moreover, we know from symmetry between $\mathbf{z}_i$ and $\mathbf{z}_i'$ that

$$\begin{aligned}\mathbb{E}\Big[\frac{\|\nabla f(A(S_i);\mathbf{z}_i')\|_2^2}{\beta_{S,A(S_i)}}\Big] &= \mathbb{E}\Big[\frac{\|\nabla f(A(S);\mathbf{z}_i)\|_2^2}{\beta_{S_i,A(S)}}\Big], \\ \mathbb{E}\Big[\frac{\|\nabla f(A(S_i);\mathbf{z}_i)\|_2^2}{\beta_{S,A(S_i)}}\Big] &= \mathbb{E}\Big[\frac{\|\nabla f(A(S);\mathbf{z}_i')\|_2^2}{\beta_{S_i,A(S)}}\Big], \\ \mathbb{E}\Big[\frac{\|\nabla F_{S_i}(A(S_i))\|_2^2}{\beta_{S,A(S_i)}}\Big] &= \mathbb{E}\Big[\frac{\|\nabla F_{S_i}(A(S))\|_2^2}{\beta_{S_i,A(S)}}\Big].\end{aligned}$$

Since $\beta_{S,A(S_i)} \geq 0$ from its definition, dividing both sides of Eq. (VII.5) by $\beta_{S,A(S_i)}$ and taking expectations leads to

$$\begin{aligned}\mathbb{E}\Big[\frac{\|\nabla F_S(A(S_i))\|_2^2}{\beta_{S,A(S_i)}}\Big] &\leq \frac{4}{n^2}\mathbb{E}\Big[\frac{\|\nabla f(A(S);\mathbf{z}_i)\|_2^2}{\beta_{S_i,A(S)}}\Big] \\ &\quad + \frac{4}{n^2}\mathbb{E}\Big[\frac{\|\nabla f(A(S);\mathbf{z}_i')\|_2^2}{\beta_{S_i,A(S)}}\Big] + 2\mathbb{E}\Big[\frac{\|\nabla F_S(A(S))\|_2^2}{\beta_{S_i,A(S)}}\Big].\end{aligned}$$

Then it follows from Eq. (VII.4) that

$$\begin{aligned}\mathbb{E}\big[f(A(S_i);\mathbf{z}_i) - f(A(S);\mathbf{z}_i)\big] &\leq \frac{2}{n}\Big(\mathbb{E}\Big[\frac{\|\nabla f(A(S);\mathbf{z}_i)\|_2^2}{\beta_{S_i,A(S)}}\Big] \\ &\quad + \mathbb{E}\Big[\frac{\|\nabla f(A(S);\mathbf{z}_i')\|_2^2}{\beta_{S_i,A(S)}}\Big]\Big) + n\mathbb{E}\Big[\frac{\|\nabla F_S(A(S))\|_2^2}{\beta_{S_i,A(S)}}\Big].\end{aligned}$$

Taking a summation of the above inequality for $i = 1, \ldots, n$, we further obtain

$$\begin{aligned}\sum_{i=1}^n \mathbb{E}\big[f(A(S_i);\mathbf{z}_i) - f(A(S);\mathbf{z}_i)\big] &\leq n\sum_{i=1}^n \mathbb{E}\Big[\frac{\|\nabla F_S(A(S))\|_2^2}{\beta_{S_i,A(S)}}\Big] \\ + \frac{2}{n}\sum_{i=1}^n \Big(\mathbb{E}\Big[\frac{\|\nabla f(A(S);\mathbf{z}_i)\|_2^2}{\beta_{S_i,A(S)}}\Big] &+ \mathbb{E}\Big[\frac{\|\nabla f(A(S);\mathbf{z}_i')\|_2^2}{\beta_{S_i,A(S)}}\Big]\Big).\end{aligned} \tag{VII.6}$$

Therefore, it follows from Definition III.2 that A is on average ϵ_A-stable with

$$\epsilon_A = \frac{2}{n^2}\sum_{i=1}^{n}\Big(\mathbb{E}\Big[\frac{\|\nabla f(A(S);\mathbf{z}_i)\|_2^2}{\beta_{S_i,A(S)}}\Big]+\mathbb{E}\Big[\frac{\|\nabla f(A(S);\mathbf{z}_i')\|_2^2}{\beta_{S_i,A(S)}}\Big]\Big)$$
$$+\sum_{i=1}^{n}\mathbb{E}\Big[\frac{\|\nabla F_S(A(S))\|_2^2}{\beta_{S_i,A(S)}}\Big] = \frac{2}{n}\Big(\mathbb{E}\Big[\frac{\|\nabla f(A(S);\mathbf{z}_1)\|_2^2}{\beta_{S_1,A(S)}}\Big]$$
$$+\mathbb{E}\Big[\frac{\|\nabla f(A(S);\mathbf{z}_1')\|_2^2}{\beta_{S_1,A(S)}}\Big]\Big)+n\mathbb{E}\Big[\frac{\|\nabla F_S(A(S))\|_2^2}{\beta_{S_1,A(S)}}\Big], \quad \text{(VII.7)}$$

where we have used the following identities due to the symmetry of the algorithm A and the fact that the samples follow from the same distribution

$$\mathbb{E}\Big[\frac{\|\nabla f(A(S);\mathbf{z}_i)\|_2^2}{\beta_{S_i,A(S)}}\Big] = \mathbb{E}\Big[\frac{\|\nabla f(A(S);\mathbf{z}_1)\|_2^2}{\beta_{S_1,A(S)}}\Big], \ \forall i\in[n],$$
$$\mathbb{E}\Big[\frac{\|\nabla f(A(S);\mathbf{z}_i')\|_2^2}{\beta_{S_i,A(S)}}\Big] = \mathbb{E}\Big[\frac{\|\nabla f(A(S);\mathbf{z}_1')\|_2^2}{\beta_{S_1,A(S)}}\Big], \ \forall i\in[n].$$

This proves Eq. (IV.4).

We proceed to prove Eq. (IV.5) using the L_S-smoothness of F_S. In this case, note from Lemma VII.1 and nonnegativity of $\beta_{S_1,A(S)}$ that

$$\mathbb{E}\Big[\frac{\|\nabla F_S(A(S))\|_2^2}{\beta_{S_1,A(S)}}\Big] \le 2\mathbb{E}\Big[\frac{L_S(F_S(A(S))-\hat{F}_S)}{\beta_{S_1,A(S)}}\Big].$$

Plugging the above inequality back into Eq. (VII.7) gives Eq. (IV.5). Finally, the generalization bounds follow directly from Lemma III.3. The proof is now complete. □

Remark VII.2. The above proof also illustrates how the classical PL condition influences stability, as studied in [16, 17]. In particular, by Eq. (VII.2) and Definition III.2, the algorithm A is on-average ϵ_A-stable with $\epsilon_A = \sum_{i=1}^{n}\mathbb{E}\big[F_S(A(S_i)) - F_S(A(S))\big]$. For simplicity, here we consider A to be the ERM method. Under the PL condition, we have $\mathbb{E}\big[F_S(A(S_i)) - F_S(A(S))\big] \le \frac{1}{2\beta_S}\mathbb{E}[\|\nabla F_S(A(S_i))\|_2^2]$. Since $\nabla F_{S_i}(A(S_i)) = 0$ by the definition of ERM, it follows that

$$\mathbb{E}[\|\nabla F_S(A(S_i))\|_2^2] = \mathbb{E}[\|\nabla F_S(A(S_i)) - \nabla F_{S_i}(A(S_i))\|_2^2]$$
$$= \frac{1}{n^2}\mathbb{E}[\|\nabla f(A(S_i);\mathbf{z}_i) - \nabla f(A(S_i);\mathbf{z}_i')\|_2^2].$$

Consequently, we have $\epsilon_A \le \frac{1}{2n^2\beta_S}\sum_{i=1}^{n}\mathbb{E}[\|\nabla f(A(S_i);\mathbf{z}_i) - \nabla f(A(S_i);\mathbf{z}_i')\|_2^2]$, which is of order $O(\frac{1}{n\beta_S})$ and highlights how the PL parameter directly controls the stability.

Next, we provide the proof of Theorem IV.8, where we shall frequently apply the following property for an L-smooth function g and $\mathbf{w},\mathbf{w}'\in\mathcal{W}$:

$$g(\mathbf{w}) \le g(\mathbf{w}') + \langle \mathbf{w}-\mathbf{w}', \nabla g(\mathbf{w}')\rangle + \frac{L\|\mathbf{w}-\mathbf{w}'\|_2^2}{2}. \quad \text{(VII.8)}$$

Proof of Theorem IV.8. Note that $F_S(\widetilde{A}(S)) = \hat{F}_S$ from the definition of $\widetilde{A}$ in Eq. (IV.11). Hence, we can apply Eq. (IV.5) with $A = \widetilde{A}$ to get

$$\mathbb{E}\big[F(\widetilde{A}(S)) - \hat{F}_S\big] = \mathbb{E}\big[F(\widetilde{A}(S)) - F_S(\widetilde{A}(S))\big]$$
$$\le \mathbb{E}\Big[\frac{2\|\nabla f(\widetilde{A}(S);\mathbf{z}_1)\|_2^2}{n\beta_{S_1,\widetilde{A}(S)}}\Big] + \mathbb{E}\Big[\frac{2\|\nabla f(\widetilde{A}(S);\mathbf{z}_1')\|_2^2}{n\beta_{S_1,\widetilde{A}(S)}}\Big]. \quad \text{(VII.9)}$$

Moreover, note from the L-smoothness of $f(\cdot;\mathbf{z})$ and Young's inequality that

$$\|\nabla f(\widetilde{A}(S);\mathbf{z}_1)\|_2^2$$
$$= \|\nabla f(A(S);\mathbf{z}_1) + \nabla f(\widetilde{A}(S);\mathbf{z}_1) - \nabla f(A(S);\mathbf{z}_1)\|_2^2$$
$$\le 2\|\nabla f(A(S);\mathbf{z}_1)\|_2^2 + 2\|\nabla f(\widetilde{A}(S);\mathbf{z}_1) - \nabla f(A(S);\mathbf{z}_1)\|_2^2$$
$$\le 2\|\nabla f(A(S);\mathbf{z}_1)\|_2^2 + 2L^2\|\widetilde{A}(S) - A(S)\|_2^2$$

and similarly

$$\|\nabla f(\widetilde{A}(S);\mathbf{z}_1')\|_2^2 \le 2\|\nabla f(A(S);\mathbf{z}_1')\|_2^2 + 2L^2\|\widetilde{A}(S) - A(S)\|_2^2.$$

Hence, it follows from Eq. (VII.9) that

$$\mathbb{E}\big[F(\widetilde{A}(S)) - \hat{F}_S\big] \le \mathbb{E}\Big[\frac{4\|\nabla f(A(S);\mathbf{z}_1)\|_2^2}{n\beta_{S_1,\widetilde{A}(S)}}$$
$$+\frac{4\|\nabla f(A(S);\mathbf{z}_1')\|_2^2}{n\beta_{S_1,\widetilde{A}(S)}} + \frac{8L^2\|A(S)-\widetilde{A}(S)\|_2^2}{n\beta_{S_1,\widetilde{A}(S)}}\Big]. \quad \text{(VII.10)}$$

Since F is L-smooth, we have from Eq. (VII.8) and the Cauchy-Schwarz inequality that for any $\gamma > 0$

$$F(A(S)) - F(\widetilde{A}(S))$$
$$\le \langle \nabla F(\widetilde{A}(S)), A(S) - \widetilde{A}(S)\rangle + \frac{L}{2}\|A(S)-\widetilde{A}(S)\|_2^2$$
$$\le \|\nabla F(\widetilde{A}(S))\|_2\|A(S)-\widetilde{A}(S)\|_2 + \frac{L}{2}\|A(S)-\widetilde{A}(S)\|_2^2$$
$$\le \frac{1}{4\gamma}\|\nabla F(\widetilde{A}(S))\|_2^2 + \Big(\gamma+\frac{L}{2}\Big)\|A(S)-\widetilde{A}(S)\|_2^2. \quad \text{(VII.11)}$$

Recall $\mathbf{w}^* := \arg\min_{\mathbf{w}\in\mathcal{W}} F(\mathbf{w})$. Applying Lemma VII.1 with $\mathbf{w} = \widetilde{A}(S)$ implies that

$$\mathbb{E}[F(A(S)) - F(\widetilde{A}(S))]$$
$$\le \frac{L}{2\gamma}\mathbb{E}\big[F(\widetilde{A}(S)) - F(\mathbf{w}^*)\big] + \Big(\gamma+\frac{L}{2}\Big)\mathbb{E}[\|A(S)-\widetilde{A}(S)\|_2^2]$$
$$\le \frac{L}{2\gamma}\mathbb{E}\big[F(\widetilde{A}(S)) - \hat{F}_S\big] + \Big(\gamma+\frac{L}{2}\Big)\mathbb{E}[\|A(S)-\widetilde{A}(S)\|_2^2],$$

where the last inequality holds due to $\mathbb{E}[\hat{F}_S] \le \mathbb{E}[F_S(\mathbf{w}^*)] = F(\mathbf{w}^*)$. Using the above inequality, we can obtain that

$$\mathbb{E}[F(A(S)) - \hat{F}_S]$$
$$= \mathbb{E}[F(A(S)) - F(\widetilde{A}(S))] + \mathbb{E}[F(\widetilde{A}(S)) - \hat{F}_S]$$
$$\le \Big(1+\frac{L}{2\gamma}\Big)\mathbb{E}\big[F(\widetilde{A}(S)) - \hat{F}_S\big] + \Big(\gamma+\frac{L}{2}\Big)\mathbb{E}[\|A(S)-\widetilde{A}(S)\|_2^2],$$

which together with Eq. (VII.10) further gives

$$\mathbb{E}[F(A(S)) - \hat{F}_S] \le \Big(\gamma+\frac{L}{2}\Big)\mathbb{E}[\|A(S)-\widetilde{A}(S)\|_2^2]$$
$$+\Big(1+\frac{L}{2\gamma}\Big)\mathbb{E}\Big[\frac{4\|\nabla f(A(S);\mathbf{z}_1)\|_2^2 + 4\|\nabla f(A(S);\mathbf{z}_1')\|_2^2}{n\beta_{S_1,\widetilde{A}(S)}}$$
$$+\frac{8L^2\|A(S)-\widetilde{A}(S)\|_2^2}{n\beta_{S_1,\widetilde{A}(S)}}\Big].$$

Plugging $\gamma = L/2$ in the above inequality leads to

$$\mathbb{E}[F(A(S)) - \hat{F}_S] \le L\mathbb{E}[\|A(S)-\widetilde{A}(S)\|_2^2] +$$
$$8\mathbb{E}\Big[\frac{\|\nabla f(A(S);\mathbf{z}_1)\|_2^2 + \|\nabla f(A(S);\mathbf{z}_1')\|_2^2}{n\beta_{S_1,\widetilde{A}(S)}} + \frac{2L^2\|A(S)-\widetilde{A}(S)\|_2^2}{n\beta_{S_1,\widetilde{A}(S)}}\Big],$$

which proves Eq. (IV.12). Additionally, according to the definition of μ_S in Eq. (IV.10), we further get

$$\mathbb{E}[F(A(S)) - \hat{F}_S] \leq 2L\mathbb{E}\Big[\frac{F_S(A(S)) - \hat{F}_S}{\mu_S}\Big] + 8\mathbb{E}\Big[\frac{\|\nabla f(A(S);\mathbf{z}_1)\|_2^2 + \|\nabla f(A(S);\mathbf{z}_1')\|_2^2}{n\beta_{S_1,\widetilde{A}(S)}} + \frac{4L^2(F_S(A(S)) - \hat{F}_S)}{n\mu_S\beta_{S_1,\widetilde{A}(S)}}\Big],$$

which implies Eq. (IV.13). The proof is now complete. □

VIII. Conclusion

In this paper, we develop a general stability and generalization framework for optimization algorithms in the nonconvex setting. Our generalization bounds depend on an algorithm-dependent quantity that can be significantly larger than the PL parameter. Moreover, we do not impose restrictive conditions such as the PL inequality and Lipschitzness of loss functions, and thus our results apply to a broad class of learning problems. Additionally, we introduce a one-point QG parameter and derive stability bounds with improved dependence on the optimization error. Our analysis is inspired by the work [16] and relaxes several of the restrictive assumptions therein, such as the QG condition. The resulting theory is highly general and applies to algorithms that do not converge to a global or local minimizer. Instead, one only needs to combine the optimization error bounds of the algorithm with our general stability bound to obtain population risk bounds. We also demonstrate the applicability of our framework by analyzing gradient descent for both linear models and SNNs.

Several open problems remain. First, our current analysis focuses on gradient descent with known linear convergence rates established in the literature [42]; it would be valuable to extend the theory to other optimization algorithms. Second, we derive stability bounds in expectation; developing high-probability bounds would provide deeper insight into the robustness of these algorithms.

References

[1] F. Cucker and D.-X. Zhou, *Learning Theory: an Approximation Theory Viewpoint*. Cambridge University Press, 2007.

[2] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of Machine Learning*. MIT press, 2012.

[3] P. L. Bartlett, D. J. Foster, and M. J. Telgarsky, "Spectrally-normalized margin bounds for neural networks," in *Advances in Neural Information Processing Systems*, 2017, pp. 6240–6249.

[4] O. Bousquet and A. Elisseeff, "Stability and generalization," *Journal of Machine Learning Research*, vol. 2, no. Mar, pp. 499–526, 2002.

[5] M. Hardt, B. Recht, and Y. Singer, "Train faster, generalize better: Stability of stochastic gradient descent," in *International Conference on Machine Learning*, 2016, pp. 1225–1234.

[6] A. Xu and M. Raginsky, "Information-theoretic analysis of generalization capability of learning algorithms," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, pp. 2525–2534.

[7] G. Neu, G. K. Dziugaite, M. Haghifam, and D. M. Roy, "Information-theoretic generalization bounds for stochastic gradient descent," in *Conference on Learning Theory*. PMLR, 2021, pp. 3526–3545.

[8] G. K. Dziugaite and D. M. Roy, "Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data," *arXiv preprint arXiv:1703.11008*, 2017.

[9] F. Hellström, G. Durisi, B. Guedj, and M. Raginsky, "Generalization bounds: Perspectives from information theory and pac-bayes," *Foundations and Trends® in Machine Learning*, vol. 18, no. 1, pp. 1–223, 2025.

[10] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding deep learning (still) requires rethinking generalization," *Communications of the ACM*, vol. 64, no. 3, pp. 107–115, 2021.

[11] I. Kuzborskij and F. Orabona, "Stability and hypothesis transfer learning," in *International Conference on Machine Learning*. PMLR, 2013, pp. 942–950.

[12] Y. Lei and Y. Ying, "Fine-grained analysis of stability and generalization for stochastic gradient descent," in *International Conference on Machine Learning*, 2020, pp. 5809–5819.

[13] C. Liu, L. Zhu, and M. Belkin, "Loss landscapes and optimization in over-parameterized non-linear systems and neural networks," *Applied and Computational Harmonic Analysis*, vol. 59, pp. 85–116, 2022.

[14] D. Richards and I. Kuzborskij, "Stability & generalisation of gradient descent for shallow neural networks without the neural tangent kernel," *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[15] H. Taheri and C. Thrampoulidis, "Generalization and stability of interpolating neural networks with minimal width," *Journal of Machine Learning Research*, vol. 25, no. 156, pp. 1–41, 2024.

[16] Z. Charles and D. Papailiopoulos, "Stability and generalization of learning algorithms that converge to global optima," in *International Conference on Machine Learning*, 2018, pp. 744–753.

[17] Y. Lei and Y. Ying, "Sharper generalization bounds for learning with gradient-dominated objective functions," in *International Conference on Learning Representations*, 2021.

[18] H. Karimi, J. Nutini, and M. Schmidt, "Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition," in *European Conference on Machine Learning*, 2016, pp. 795–811.

[19] O. Bousquet, Y. Klochkov, and N. Zhivotovskiy, "Sharper bounds for uniformly stable algorithms," in *Conference on Learning Theory*, 2020, pp. 610–626.

[20] V. Feldman and J. Vondrak, "High probability generalization bounds for uniformly stable algorithms with nearly optimal rate," in *Conference on Learning Theory*, 2019, pp. 1270–1279.

[21] S. Li, B. Zhu, and Y. Liu, "Algorithmic stability unleashed: generalization bounds with unbounded losses," in *International Conference on Machine Learning*, 2024.

[22] S. Shalev-Shwartz, O. Shamir, N. Srebro, and K. Sridharan, "Learnability, stability and uniform convergence," *Journal of Machine Learning Research*, vol. 11, no. Oct, pp. 2635–2670, 2010.

[23] R. Bassily, V. Feldman, C. Guzmán, and K. Talwar, "Stability of stochastic gradient descent on nonsmooth convex losses," *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[24] J. Li, X. Luo, and M. Qiao, "On generalization error bounds of noisy gradient methods for non-convex learning," in *International Conference on Learning Representations*, 2020.

[25] M. Schliserman and T. Koren, "Stability vs implicit bias of gradient methods on separable data and beyond," in *Conference on Learning Theory*, 2022, pp. 3380–3394.

[26] I. Kuzborskij and C. Lampert, "Data-dependent stability of stochastic gradient descent," in *International Conference on Machine Learning*, 2018, pp. 2820–2829.

[27] I. Amir, T. Koren, and R. Livni, "Sgd generalizes better than gd (and regularization doesn't help)," in *Conference on Learning Theory*. PMLR, 2021, pp. 63–92.

[28] P. Zhang, J. Teng, and J. Zhang, "Generalization lower bounds for gd and sgd in smooth stochastic convex optimization," in *International Conference on Artificial Intelligence and Statistics*, 2025.

[29] K. Nikolakakis, F. Haddadpour, A. Karbasi, and D. Kalogerias, "Beyond lipschitz: Sharp generalization and excess risk bounds for full-batch gd," in *The Eleventh International Conference on Learning Representations*, 2023.

[30] H. Taheri, C. Thrampoulidis, and A. Mazumdar, "Sharper guarantees for learning neural network classifiers with gradient methods," in *The Thirteenth International Conference on Learning Representations*, 2025.

[31] W. Ma, Y. Sui, J. Teng, B. Wang, J. Xu, and J. Yang, "Generalization bounds of stochastic gradient descent in homogeneous neural networks," *arXiv preprint arXiv:2602.22936*, 2026.

[32] Y. Wang and R. Arora, "On the stability and generalization of meta-learning," in *Advances in Neural Information Processing Systems*, 2024.

[33] A. Maurer, "Algorithmic stability and meta-learning," *Journal of Machine Learning Research*, vol. 6, no. Jun, pp. 967–994, 2005.

[34] J. Chen, H. Chen, X. Jiang, B. Gu, W. Li, T. Gong, and F. Zheng, "On the stability and generalization of triplet learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 6, 2023, pp. 7033–7041.

[35] K. Nikolakakis, F. Haddadpour, D. Kalogerias, and A. Karbasi, "Black-box generalization: Stability of zeroth-order learning," *Advances in Neural Information Processing Systems*, vol. 35, pp. 31 525–31 541, 2022.

[36] X. Deng, L. Shen, S. Li, T. Sun, D. Li, and D. Tao, "Towards understanding the generalizability of delayed stochastic gradient descent," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[37] S. Zeng and Y. Lei, "Stochastic gradient methods: Bias, stability and generalization," *Journal of Machine Learning Research*, vol. 27, no. 6, pp. 1–55, 2026.

[38] P. Wen, Q. Xu, Z. Yang, Y. He, and Q. Huang, "Algorithm-dependent generalization of AUPRC optimization: Theory and algorithm," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 7, pp. 5062–5079, 2024.

[39] Y. Sun, L. Shen, and D. Tao, "Towards understanding generalization and stability gaps between centralized and decentralized federated learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–12, 2026.

[40] O. Bousquet and L. Bottou, "The tradeoffs of large scale learning," in *Advances in Neural Information Processing Systems*, 2008, pp. 161–168.

[41] L. Bottou, F. E. Curtis, and J. Nocedal, "Optimization methods for large-scale machine learning," *SIAM Review*, vol. 60, no. 2, pp. 223–311, 2018.

[42] S. Oymak and M. Soltanolkotabi, "Overparameterized nonlinear learning: Gradient descent takes the shortest path?" in *International Conference on Machine Learning*. PMLR, 2019, pp. 4951–4960.

[43] A. Gonen and S. Shalev-Shwartz, "Average stability is invariant to data preconditioning. implications to exp-concave empirical risk minimization," *Journal of Machine Learning Research*, vol. 18, no. 222, pp. 1–13, 2018.

[44] S. Arora, S. Du, W. Hu, Z. Li, and R. Wang, "Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks," in *International Conference on Machine Learning*, 2019, pp. 322–332.

[45] S. Oymak and M. Soltanolkotabi, "Towards moderate overparameterization: global convergence guarantees for training shallow neural networks," *IEEE Journal on Selected Areas in Information Theory*, 2020.

[46] M. Soltanolkotabi, A. Javanmard, and J. D. Lee, "Theoretical insights into the optimization landscape of over-parameterized shallow neural networks," *IEEE Transactions on Information Theory*, vol. 65, no. 2, pp. 742–769, 2018.

[47] Y. Lei, R. Jin, and Y. Ying, "Stability and generalization analysis of gradient methods for shallow neural networks," in *Advances in Neural Information Processing Systems*, vol. 35, 2022, pp. 38 557–38 570.

[48] P. Deora, R. Ghaderi, H. Taheri, and C. Thrampoulidis, "On the optimization and generalization of multi-head attention," *Transactions on Machine Learning Research*, 2024.

[49] C.-C. Chang and C.-J. Lin, "Libsvm: a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, p. 27, 2011.

[50] N. Srebro, K. Sridharan, and A. Tewari, "Smoothness, low noise and fast rates," in *Advances in Neural Information Processing Systems*, 2010, pp. 2199–2207.

Yunwen Lei received the Ph.D. degree from the Wuhan University, Wuhan, China, in 2014. He is currently an Assistant Professor with the Department of Mathematics, The University of Hong Kong. His research interests include machine learning, data science, learning theory, and stochastic optimization.

Zimeng Wang obtained the Ph.D. degree from The University of Hong Kong in 2025. He is currently a postdoctoral researcher with the Department of Mathematics and Mathematical Statistics, Umeå University. His research interests include optimization theory (especially stochastic optimization), statistical learning theory, and their interactions.

Xiaoming Yuan is currently a Professor with the Department of Mathematics, The University of Hong Kong. His research interests include optimization, scientific machine learning, artificial intelligence, and cloud computing.