Data-Dependent Stability of Stochastic Gradient Descent

Ilja Kuzborskij
EPFL
Switzerland
ilja.kuzborskij@gmail.com

Christoph H. Lampert
IST Austria
Klosterneuburg, Austria
chl@ist.ac.at

February 19, 2018

Abstract

We establish a data-dependent notion of algorithmic stability for Stochastic Gradient Descent (SGD), and employ it to develop novel generalization bounds. This is in contrast to previous distribution-free algorithmic stability results for SGD which depend on the worst-case constants. By virtue of the data-dependent argument, our bounds provide new insights into learning with SGD on convex and non-convex problems. In the convex case, we show that the bound on the generalization error depends on the risk at the initialization point. In the non-convex case, we prove that the expected curvature of the objective function around the initialization point has crucial influence on the generalization error. In both cases, our results suggest a simple data-driven strategy to stabilize SGD by pre-screening its initialization. As a corollary, our results allow us to show optimistic generalization bounds that exhibit fast convergence rates for SGD subject to a vanishing empirical risk and low noise of stochastic gradient.

1 Introduction

Stochastic gradient descent (SGD) has become one of the workhorses of modern machine learning. In particular, it is the optimization method of choice for training highly complex and non-convex models, such as neural networks. When it was observed that these models generalize better (suffer less from overfitting) than classical machine learning theory suggests, a large theoretical interest emerged to explain this phenomenon. Given that SGD at best finds a local minimum of the non-convex objective function, it has been argued that all such minima might be equally good. However, at the same time, a large body of empirical work and tricks of trade, such as *early stopping*, suggests that in practice one might not even reach a minimum, yet nevertheless observes excellent performance.

In this work we follow an alternative route that aims to *directly* analyze the generalization ability of SGD by studying how sensitive it is to small perturbations in the training set. This is known as *algorithmic stability* approach [4] and was used recently [15] to establish generalization bounds for both convex and non-convex learning settings. To do so they employed a rather restrictive notion of stability that does not depend on the data, but captures only intrinsic characteristics of the learning algorithm and global properties of the objective function. Consequently, their analysis results in worst-case guarantees that in some cases tend to be too pessimistic. As recently pointed out in [34], *deep learning* might indeed be such a case, as this notion of stability is insufficient to give deeper theoretical insights, and a less restrictive one is desirable.

As our main contribution in this work we establish that a data-dependent notion of algorithmic stability, very similar to the *On-Average Stability* [32], holds for SGD when applied to convex as well as non-convex learning problems. As a consequence we obtain new generalization bounds that depend on the data-generating distribution and the initialization point of an algorithm. For convex loss functions, the bound on the generalization error is essentially multiplicative in the risk at the initialization point when noise of stochastic gradient is not too high. For the non-convex loss functions, besides the risk, it is also critically controlled by the expected second-order information about the objective function at the initialization point. We further corroborate our findings empirically and show that, indeed, the data-dependent generalization bound is tighter than the worst-case counterpart on non-convex objective functions. Finally, the nature of the data-dependent bounds allows us to state *optimistic* bounds that switch to the faster rate of convergence subject to the vanishing empirical risk.

In particular, our findings justify the intuition that SGD is more stable in less curved areas of the objective function and link it to the generalization ability. This also backs up numerous empirical findings in the deep learning literature that solutions with low generalization error occur in less curved regions. At the same time, in pessimistic scenarios, our bounds are no worse than those of [15].

Finally, we exemplify an application of our bounds, and propose a simple yet principled *transfer learning* scheme for the convex and non-convex case, which is guaranteed to transfer from the best source of information. In addition, this approach can also be used to select a good initialization given a number of random starting positions. This is a theoretically sound alternative to the purely random commonly used in non-convex learning.

The rest of the paper is organized as follows. We revisit the connection between stability and generalization of SGD in Section 3 and introduce a data-dependent notion of stability in Section 4. We state the main results in Section 5, in particular, Theorem 3 for the convex case, and Theorem 4 for the non-convex one. Next we demonstrate empirically that the bound shown in Theorem 4 is tighter than the worst-case one in Section 5.2.1. Finally, we suggest application of these bounds by showcasing principled transfer learning approaches in Section 5.3, and we conclude in Section 6.

2 Related Work

Algorithmic stability has been a topic of interest in learning theory for a long time, however, the modern approach on the relationship between stability and generalization goes back to the milestone work of [4]. They analyzed several notions of stability, which fall into two categories: distribution-free and distribution-dependent ones. The first category is usually called *uniform* stability and focuses on the intrinsic stability properties of an algorithm without regard to the data-generating distribution. Uniform stability was used to analyze many algorithms, including regularized Empirical Risk Minimization (ERM) [4], randomized aggregation schemes [9], and recently SGD by [15, 23], and [29]. Despite the fact that uniform stability has been shown to be sufficient to guarantee learnability, it can be too pessimistic, resulting in worst-case rates.

In this work we are interested in the data-dependent behavior of SGD, thus the emphasis will fall on the distribution-dependent notion of stability, known as *on-average* stability, explored throughly in [32]. The attractive quality of this less restrictive stability type is that the resulting bounds are controlled by how stable the algorithm is under the data-generating distribution. For instance, in [4] and [8], the on-average stability is related to the variance of an estimator. In [31, Sec. 13], the authors show risk bounds that depend on the expected empirical risk of a solution to the regularized ERM. In turn, one can exploit this fact to state improved *optimistic* risk bounds, for instance, ones that exhibit *fast-rate* regimes [19, 13], or even to design enhanced algorithms that minimize these bounds in a data-driven way, e.g. by exploiting side information as in transfer [20, 2] and metric learning [28]. Here, we mainly focus on the later direction in the context of SGD: how stable is SGD under the data-generating distribution given an initialization point? We also touch the former direction by taking advantage of our data-driven analysis and show optimistic bounds as a corollary.

We will study the on-average stability of SGD for both convex and non-convex loss functions. In the convex setting, we will relate stability to the risk at the initialization point, while previous data-driven stability arguments usually consider minimizers of convex ERM rather than a stochastic approximation [31, 19]. Beside convex problems, our work also covers the generalization ability of SGD on non-convex problems. Here, we borrow techniques of [15] and extend them to the distribution-dependent setting. That said, while bounds of [15] are stated in terms of worst-case quantities, ours reveal new connections to the data-dependent second-order information. These new insights also partially justify empirical observations in deep learning about the link between the curvature and the generalization error [16, 18, 5]. At the same time, our work is an alternative to the theoretical studies of neural network objective functions [6, 17], as we focus on the direct connection between the generalization and the curvature.

In this light, our work is also related to non-convex optimization by SGD. Literature on this subject typically studies rates of convergence to the stationary points [12, 1, 30], and ways to avoid saddles [11, 22]. However, unlike these works, and similarly to [15], we are interested in the generalization ability of SGD, and thanks to the stability approach, involvement of stationary points in our analysis is not necessary.

Finally, we propose an example application of our findings in Transfer Learning (TL). For instance, by controlling the stability bound in a data-driven way, one can choose an initialization that leads to improved generalization. This is related to TL where one transfers from pre-trained models [21, 33, 27, 2], especially popular in deep learning due to its data-demanding nature [10]. Literature on this topic is mostly focused on the ERM setting and PAC-bounds, while our analysis of SGD yields such guarantees as a corollary.

3 Stability of Stochastic Gradient Descent

First, we introduce definitions used in the rest of the paper.

3.1 Definitions

We will denote with small and capital bold letters respectively column vectors and matrices, e.g., $\boldsymbol{a} = [a_1, a_2, \dots, a_d]^T \in \mathbb{R}^d$ and $\boldsymbol{A} \in \mathbb{R}^{d_1 \times d_2}$, $\|\boldsymbol{a}\|$ is understood as a Euclidean norm and $\|\boldsymbol{A}\|_2$ as the spectral norm. We denote enumeration by $[n] = \{1, \dots, n\}$ for $n \in \mathbb{N}$.

We indicate an example space by $\mathcal{Z}$ and its member by $z \in \mathcal{Z}$. For instance, in a supervised setting $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$, such that $\mathcal{X}$ is the input and $\mathcal{Y}$ is the output space of a learning problem. We assume that training and testing examples are drawn iid from a probability distribution $\mathcal{D}$ over $\mathcal{Z}$. In particular, we will denote the training set as $S = \{z_i\}_{i=1}^m \sim \mathcal{D}^m$.

For a parameter space $\mathcal{H}$, we define a learning algorithm as a map $A : \mathcal{Z}^m \mapsto \mathcal{H}$ and for brevity we will use the notation $A_S = A(S)$. In the following we assume that $\mathcal{H} \subseteq \mathbb{R}^d$. To measure the accuracy of a learning algorithm A, we have a *loss* function $f(\boldsymbol{w}, z)$, which measures the cost incurred by predicting with parameters $\boldsymbol{w} \in \mathcal{H}$ on an example z. The *risk* of $\boldsymbol{w}$, with respect to the distribution $\mathcal{D}$, and the *empirical risk* given a training set S are defined as

$$R(\boldsymbol{w}) := \mathop{\mathbb{E}}_{z\sim\mathcal{D}}[f(\boldsymbol{w}, z)], \quad \text{and} \quad \widehat{R}_S(\boldsymbol{w}) := \frac{1}{m}\sum_{i=1}^{m} f(\boldsymbol{w}, z_i) \, .$$

Finally, define $R^\star := \inf_{\boldsymbol{w}\in\mathcal{H}} R(\boldsymbol{w})$.

3.2 Uniform Stability and Generalization

On an intuitive level, a learning algorithm is said to be *stable* whenever a small perturbation in the training set does not affect its outcome too much. Of course, there is a number of ways to formalize the perturbation and the extent of the change in the outcome, and we will discuss some of them below. The most important consequence of a stable algorithm is that it *generalizes* from the training set to the unseen data sampled from the same distribution. In other words, the difference between the risk $R(A_S)$ and the empirical risk $\widehat{R}_S(A_S)$ of the algorithm's output is controlled by the quantity that captures how stable the algorithm is. So, to observe good performance, or a decreasing true risk, we must have a stable algorithm *and* decreasing empirical risk (training error), which usually comes by design of the algorithm. In this work we focus on the stability of the Stochastic Gradient Descent (SGD) algorithm, and thus, as a consequence, we study its generalization ability.

Recently, [15] used a stability argument to prove generalization bounds for learning with SGD. Specifically, the authors extended the notion of the *uniform stability* originally proposed by [4], to accommodate randomized algorithms.

Definition 1 (Uniform stability). *A randomized algorithm A is ϵ-uniformly stable if for all datasets $S, S^{(i)} \in \mathcal{Z}^m$ such that S and $S^{(i)}$ differ in the i-th example, we have*

$$\sup_{z \in \mathcal{Z}, i \in [m]} \left\{ \mathop{\mathbb{E}}_{A} \left[f(A_S, z) - f(A_{S^{(i)}}, z) \right] \right\} \leqslant \epsilon \ .$$

Since SGD is a randomized algorithm, we have to cope with two sources of randomness: the data-generating process and the randomization of the algorithm A itself, hence we have statements in expectation. The following theorem of [15] shows that the uniform stability implies generalization in expectation.

Theorem 1. *Let A be ϵ-uniformly stable. Then,*

$$\left| \mathop{\mathbb{E}}_{S,A} \left[\widehat{R}_S(A_S) - R(A_S) \right] \right| \leqslant \epsilon \ .$$

Thus it suffices to characterize the uniform stability of an algorithm to state a generalization bound. In particular, [15] showed generalization bounds for SGD under different assumptions on the loss function f. Despite that these results hold in expectation, other forms of generalization bounds, such as high-probability ones, can be derived from the above [32].

Apart from SGD, uniform stability has been used before to prove generalization bounds for many learning algorithms [4]. However, these bounds typically suggest worst-case generalization rates, and rather reflect intrinsic stability properties of an algorithm. In other words, uniform stability is oblivious to the data-generating process and any other side information, which might reveal scenarios where generalization occurs at a faster rate. In turn, these insights could motivate the design of improved learning algorithms. In the following we address some limitations of analysis through uniform stability by using a less restrictive notion of stability. We extend the setting of [15] by proving data-dependent stability bounds for convex and non-convex loss functions. In addition, we also take into account the initialization point of an algorithm as a form of supplementary information, and we dedicate special attention to its interplay with the data-generating distribution. Finally, we discuss situations where one can explicitly control the stability of SGD in a data-dependent way.

4 Data-dependent Stability Bounds for SGD

In this section we describe a notion of data-dependent algorithmic stability, that allows us to state generalization bounds which depend not only on the properties of the learning algorithm, but also on the additional parameters of the algorithm. We indicate such additional parameters by θ, and therefore we denote stability as a function $\epsilon(\theta)$. In particular, in the following we will be interested in scenarios where θ describes the data-generating distribution and the initialization point of SGD.

Definition 2 (On-Average stability). *A randomized algorithm A is $\epsilon(\theta)$-on-average stable if it is true that*

$$\sup_{i \in [m]} \left\{ \mathop{\mathbb{E}}_{A} \mathop{\mathbb{E}}_{S,z} \left[f(A_S, z) - f(A_{S^{(i)}}, z) \right] \right\} \leqslant \epsilon(\theta) \ ,$$

where $S \overset{iid}{\sim} \mathcal{D}^m$ and $S^{(i)}$ is its copy with i-th example replaced by $z \overset{iid}{\sim} \mathcal{D}$.

Our definition of on-average stability resembles the notion introduced by [32]. The difference lies in the fact that we take supremum over index of replaced example. A similar notion was also used by [4] and later by [9] for analysis of a randomized aggregation schemes, however their definition involves absolute difference of losses. The dependence on θ also bears similarity to recent work of [23], however, there, it is used in the context of uniform stability. The following theorem shows that on-average - stable random algorithm is guaranteed to generalize in expectation.

Theorem 2. *Let an algorithm A be $\epsilon(\theta)$-on-average stable. Then,*

$$\mathop{\mathbb{E}}_{S} \mathop{\mathbb{E}}_{A} \left[R(A_S) - \widehat{R}_S(A_S) \right] \leqslant \epsilon(\theta) \ .$$

5 Main Results

Before presenting our main results in this section, we discuss algorithmic details and assumptions. We will study the following variant of SGD: given a training set $S = \{z_i\}_{i=1}^m \overset{\text{iid}}{\sim} \mathcal{D}^m$, step sizes $\{\alpha_t\}_{t=1}^T$, random indices $I = \{j_t\}_{t=1}^T$, and an initialization point $\boldsymbol{w}_1$, perform updates

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \alpha_t \nabla f(\boldsymbol{w}_t, z_{j_t})$$

for $T \leqslant m$ steps. Moreover we will use the notation $\boldsymbol{w}_{S,t}$ to indicate the output of SGD ran on a training set S, at step t. We assume that the indices in I are sampled from the uniform distribution over $[m]$ *without* replacement, and that this is the only source of randomness for SGD. In practice this corresponds to permuting the training set before making a pass through it, as it is commonly done in practical applications. We also assume that the variance of stochastic gradients obeys

$$\mathop{\mathbb{E}}_{S,z}\left[\|\nabla f(\boldsymbol{w}_{S,t}, z) - \nabla R(\boldsymbol{w}_{S,t})\|^2\right] \leqslant \sigma^2 \quad \forall t \in [T] .$$

Next, we introduce statements about the loss functions f used in the following.

Definition 3 (Lipschitz f)**.** *A loss function f is L-Lipschitz if $\|\nabla f(\boldsymbol{w}, z)\| \leqslant L$, $\forall \boldsymbol{w} \in \mathcal{H}$ and $\forall z \in \mathcal{Z}$. Note that this also implies that $|f(\boldsymbol{w}, z) - f(\boldsymbol{v}, z)| \leqslant L\|\boldsymbol{w} - \boldsymbol{v}\|$.*

Definition 4 (Smooth f)**.** *A loss function is β-smooth if $\forall \boldsymbol{w}, \boldsymbol{v} \in \mathcal{H}$ and $\forall z \in \mathcal{Z}$, $\|\nabla f(\boldsymbol{w}, z) - \nabla f(\boldsymbol{v}, z)\| \leqslant \beta\|\boldsymbol{w} - \boldsymbol{v}\|$, which also implies $f(\boldsymbol{w}, z) - f(\boldsymbol{v}, z) \leqslant \nabla f(\boldsymbol{v}, z)^\top(\boldsymbol{w} - \boldsymbol{v}) + \frac{\beta}{2}\|\boldsymbol{w} - \boldsymbol{v}\|^2$.*

Definition 5 (Lipschitz Hessians)**.** *A loss function f has a ρ-Lipschitz Hessian if $\forall \boldsymbol{w}, \boldsymbol{v} \in \mathcal{H}$ and $\forall z \in \mathcal{Z}$, $\|\nabla^2 f(\boldsymbol{w}, z) - \nabla^2 f(\boldsymbol{v}, z)\|_2 \leqslant \rho\|\boldsymbol{w} - \boldsymbol{v}\|$.*

The last condition is occasionally used in analysis of SGD [11] and holds whenever f has a bounded third derivative. All presented theorems assume that the loss function used by SGD is non-negative, Lipschitz, and β-smooth. Examples of such commonly used loss functions are the logistic/softmax losses and neural networks with sigmoid activations. Convexity of loss functions or Lipschitzness of Hessians will only be required for some results, and we will denote it explicitly when necessary. Proofs for all the statements in this section are given in the supplementary material.

5.1 Convex Losses

First, we present a new and data-dependent stability result for convex losses.

Theorem 3. *Assume that f is convex, and that SGD's step sizes satisfy $\alpha_t = \frac{c}{\sqrt{t}} \leqslant \frac{1}{\beta}$, $\forall t \in [T]$. Then SGD is $\epsilon(\mathcal{D}, \boldsymbol{w}_1)$-on-average stable with*

$$\epsilon(\mathcal{D}, \boldsymbol{w}_1) = \mathcal{O}\left(\sqrt{c\,(R(\boldsymbol{w}_1) - R^\star)} \cdot \frac{\sqrt[4]{T}}{m} + c\sigma\frac{\sqrt{T}}{m}\right) .$$

Under the same assumptions, taking step size of order $\mathcal{O}(1/\sqrt{t})$, [15] showed a uniform stability bound $\epsilon = \mathcal{O}(\sqrt{T/m})$. Our bound differs since it involves a multiplicative risk at the initialization point. Thus, our bound corroborates the intuition that whenever we start at a good location of the objective function, the algorithm is more stable and thus generalizes better. However, this is only the case, whenever the variance of stochastic gradient σ^2 is not too large. In the extreme case, deterministic case, and of $R(\boldsymbol{w}_1) = 0$, the theorem confirms that SGD, in expectation, does not need to make any updates and is therefore perfectly stable. On the other hand, when the variance σ^2 is large enough to make the second summand in Theorem 3 dominant, the bound does not offer improvement compared to [15]. Note, that a result of this type cannot be obtained through the more restrictive uniform stability, precisely because such bounds on the stability must hold even for a worst-case choice of data distribution and initialization.

In contrast, the notion of stability we employ depends on the data-generating distribution, which allowed us to introduce dependency on the risk.

Furthermore, consider that we start at arbitrary location $\boldsymbol{w}_1$: assuming that the loss function is bounded for a concrete $\mathcal{H}$ and $\mathcal{Z}$, the rate of our bound up to a constant is no worse than that of [15]. Finally, one can always tighten this result by taking the minimum of two bounds.

5.2 Non-convex Losses

Now we state a new stability result for non-convex losses.

Theorem 4. *Assume that $f(\cdot, z) \in [0,1]$ and has a ρ-Lipschitz Hessian, and that step sizes of a form $\alpha_t = \frac{c}{t}$ satisfy $c \leqslant \min\left\{\frac{1}{\beta}, \frac{1}{4(2\beta\ln(T))^2}\right\}$. Then SGD is $\epsilon(\mathcal{D}, \boldsymbol{w}_1)$-on-average stable with*

$$\epsilon(\mathcal{D}, \boldsymbol{w}_1) \leqslant \frac{1 + \frac{1}{c\gamma}}{m} \left(2cL^2\right)^{\frac{1}{1+c\gamma}} \left(\mathop{\mathbb{E}}_{S,A}\left[R(A_S)\right] \cdot T\right)^{\frac{c\gamma}{1+c\gamma}} , \tag{1}$$

where

$$\gamma := \tilde{\mathcal{O}}\left(\min\left\{\beta,\ \mathop{\mathbb{E}}_{z}\left[\left\|\nabla^2 f(\boldsymbol{w}_1, z)\right\|_2\right] + \Delta^\star_{1,\sigma^2}\right\}\right) , \tag{2}$$

$$\Delta^\star_{1,\sigma^2} := \rho\left(c\sigma + \sqrt{c\left(R(\boldsymbol{w}_1) - R^\star\right)}\right) .$$

In particular, γ characterizes how the curvature at the initialization point affects stability, and hence the generalization error of SGD. Since γ heavily affects the rate of convergence in (1), and in most situations smaller γ yields higher stability, we now look at a few cases of its behavior. Consider a regime such that γ is of the order $\tilde{\Theta}\left(\mathbb{E}[\|\nabla^2 f(\boldsymbol{w}_1, z)\|_2] + \sqrt{R(\boldsymbol{w}_1)} + \sigma\right)$, or in other words, that stability is controlled by the curvature, the risk of the initialization point $\boldsymbol{w}_1$, and the variance of the stochastic gradient σ^2. This suggests that starting from a point in a less curved region with low risk should yield higher stability, and therefore as predicted by our theory, allow for faster generalization. In addition, we observe that the considered stability regime offers a principled way to pre-screen a good initialization point in practice, by choosing the one that minimizes spectral norm of the Hessian and the risk.

Next, we focus on a more specific case. Suppose that we choose a step size $\alpha_t = \frac{c}{t}$ such that $\gamma = \tilde{\Theta}\left(\mathbb{E}[\|\nabla^2 f(\boldsymbol{w}_1, z)\|_2]\right)$, yet not too small, so that the empirical risk can still be decreased. Then, stability is dominated by the curvature around $\boldsymbol{w}_1$. Indeed, lower generalization errors on non-convex problems, such as training deep neural networks, have been observed empirically when SGD is actively guided [16, 14, 5] or converges to solutions with low curvature [18]. However, to the best of our knowledge, Theorem 4 is the first to establish a theoretical link between the curvature of the loss function and the generalization ability of SGD in a data-dependent sense.

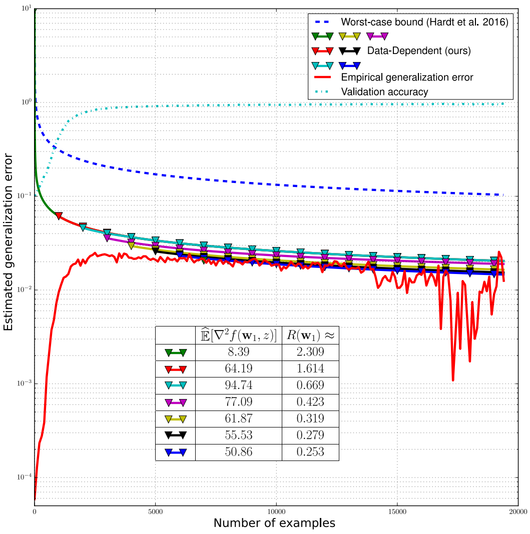
Theorem 4 immediately implies following statement that further reinforces the effect of the initialization point on the generalization error, assuming that $\mathbb{E}_S[R(A_S)] \leqslant R(\boldsymbol{w}_1)$.

Corollary 1. *Under conditions of Theorem 4 we have that SGD is $\epsilon(\mathcal{D}, \boldsymbol{w}_1)$-on-average stable with*

$$\epsilon(\mathcal{D}, \boldsymbol{w}_1) = \mathcal{O}\left(\frac{1 + \frac{1}{c\gamma}}{m} \left(R(\boldsymbol{w}_1) \cdot T\right)^{\frac{c\gamma}{1+c\gamma}}\right) . \tag{3}$$

We take a moment to discuss the role of the risk term in $(R(\boldsymbol{w}_1) \cdot T)^{\frac{c\gamma}{1+c\gamma}}$. Observe that $\epsilon(\mathcal{D}, \boldsymbol{w}_1) \to 0$ as $R(\boldsymbol{w}_1) \to 0$, in other words, the generalization error approaches zero as the risk of the initialization point vanishes. This is an intuitive behavior, however, uniform stability does not capture this due to its distribution-free nature. Finally, we note that [15, Theorem 3.8] showed a bound similar to (1), however, in place of γ their bound has a Lipschitz constant of the gradient. The crucial difference lies in term γ which is now not merely a Lipschitz constant, but rather depends on the data-generating distribution and

Figure 1: Empirical tightness of data-dependent and uniform generalization bounds evaluated by training a convolutional neural network.



initialization point of SGD. We compare to their bound by considering the worst case scenario, namely, that SGD is initialized in a point with high curvature, or altogether, that the objective function is highly curved everywhere. Then, at least our bound is no worse than the one of [15], since $\gamma \leqslant \beta$.

Theorem 4 also allows us to prove an optimistic generalization bound for learning with SGD on non-convex objectives.

Corollary 2. *Under conditions of Theorem 4 we have that the output of SGD obeys*

$$\mathop{\mathbb{E}}_{S,A}\left[R(A_S) - \widehat{R}_S(A_S)\right] = \mathcal{O}\left(\frac{1 + \frac{1}{c\gamma}}{m} \cdot \max\left\{\left(\mathop{\mathbb{E}}_{S,A}\left[\widehat{R}_S(A_S)\right] \cdot T\right)^{\frac{c\gamma}{1+c\gamma}}, \left(\frac{T}{m}\right)^{c\gamma}\right\}\right) .$$

An important consequence of Corollary 2, is that for a vanishing expected empirical risk, in particular for $\mathbb{E}_{S,A}[\widehat{R}_S(A_S)] = \mathcal{O}\left(\frac{T^{c\gamma}}{m^{1+c\gamma}}\right)$, the generalization error behaves as $\mathcal{O}\left(\frac{T^{c\gamma}}{m^{1+c\gamma}}\right)$. Considering the full pass, that is $m = \mathcal{O}(T)$, we have an optimistic generalization error of order $\mathcal{O}\left(1/m\right)$ instead of $\mathcal{O}(m^{-\frac{1}{1+c\gamma}})$. We note that PAC bounds with similar optimistic message (although not directly comparable), but without curvature information can also be obtained through empirical Bernstein bounds as in [24]. However, a PAC bound does not suggest a way to minimize non-convex empirical risk in general, where, on the other hand, SGD is known to work reasonably well.

5.2.1 Tightness of Non-convex Bounds

Next we empirically assess the tightness of our non-convex generalization bounds on real data. In the following experiment we train a neural network with three convolutional layers interlaced with max-pooling, followed by the fully connected layer with 16 units, on the MNIST dataset. This totals in a model with 18K parameters. Figure 1 compares our data-dependent bound (1) to the distribution-free one of [15, Theorem 3.8]. As as a reference we also include an empirical estimate of the generalization error taken as an absolute difference of the validation and training average losses. Since our bound also depends on the initialization point, we plot (1) for multiple “warm-starts”, ie.with SGD initialized from a pre-trained position. We consider 7 such warm-starts at every 200 steps, and report data-dependent quantities used to compute (1) just beneath the graph. Our first observation is that, clearly, the data-dependent bound gives tighter estimate, by roughly one order of magnitude. Second, simulating start from a pre-trained position suggests even tighter estimates: we suspect that this is due to decreasing validation error which is used as an empirical estimate for $R(\boldsymbol{w}_1)$ which affects bound (1).

We compute an empirical estimate of the expected Hessian spectral norm by the power iteration method using an efficient Hessian-vector multiplication method [26]. Since bounds depend on constants L, β, and ρ, we estimate them by tracking maximal values of the gradient and Hessian norms throughout optimization. We compute bounds with estimates $\widehat{L} = 78.72$, $\widehat{\beta} = 1692.28$, $\widehat{\rho} = 3823.73$, and $c = 10^{-3}$.

5.3 Application to Transfer Learning

One example application of data-dependent bounds presented before lies in *Transfer Learning (TL)*, where we are interested in achieving faster generalization on a *target* task by exploiting side information that originates from different but related *source* tasks. The literature on TL explored many ways to do so, and here we will focus on the one that is most compatible with our bounds. More formally, suppose that the *target* task at hand is characterized by a joint probability distribution $\mathcal{D}$, and as before we have a training set $S \overset{\text{iid}}{\sim} \mathcal{D}^m$. Some TL approaches also assume access to the data sampled from the distributions associated with the *source* tasks. Here we follow a conservative approach – instead of the source data, we receive a set of *source* hypotheses $\{\boldsymbol{w}_k^{\text{src}}\}_{k=1}^K \subset \mathcal{H}$, trained on the source tasks. The goal of a learner is to come up with a target hypothesis, which in the optimistic scenario generalizes better by relying on source hypotheses. In the TL literature this is known as Hypothesis Transfer Learning (HTL) [21], that is, we transfer from the source hypotheses which act as a proxy to the source tasks and the risk $R(\boldsymbol{w}_k^{\text{src}})$ quantifies how much source and target tasks are related. In the following we will consider SGD for HTL, where the source hypotheses act as initialization points. First, consider learning with convex losses: Theorem 3 depends on $R(\boldsymbol{w}_1)$, thus it immediately quantifies the relatedness of source and target tasks. So it is enough to pick the point that minimizes the stability bound to transfer from the most related source. Then, bounding $R(\boldsymbol{w}_k^{\text{src}})$ by $\widehat{R}_S(\boldsymbol{w}_k^{\text{src}})$ through Hoeffding bound along with union bound gives with high probability that

$$\min_{k \in [K]} \epsilon(\mathcal{D}, \boldsymbol{w}_k^{\text{src}}) \leqslant \min_{k \in [K]} \mathcal{O}\left(\widehat{R}_S(\boldsymbol{w}_k^{\text{src}}) + \sqrt{\frac{\log(K)}{m}} \right) .$$

Hence, the most related source is the one that simply minimizes empirical risk. Similar conclusions where drawn in HTL literature, albeit in the context of ERM. Matters are slightly more complicated in the non-convex case. We take a similar approach, however, now we minimize stability bound (3), and for the sake of simplicity assume that we make a full pass over the data, so $T = m$. Minimizing the following empirical upper bound select the best source.

Proposition 1. *Let* $\widehat{\gamma}_k^{\pm} = \Theta\Big(\frac{1}{m}\sum_{i=1}^m \|\nabla^2 f(\boldsymbol{w}_k^{src}, z_i)\|_2 + \sqrt{\widehat{R}_S(\boldsymbol{w}_k^{src}) \pm \sqrt[4]{\log(K)/m}}\Big)$. *Then with high probability the generalization error of* $\boldsymbol{w}_k^{src}$ *is bounded by*

$$\min_{k \in [K]} \mathcal{O}\left(\left(1 + \frac{1}{c\widehat{\gamma}_k^-}\right) \widehat{R}_S(\boldsymbol{w}_k^{src})^{\frac{c\widehat{\gamma}_k^+}{1+c\widehat{\gamma}_k^+}} \cdot \frac{\sqrt{\log(K)}}{m^{\frac{1}{1+c\widehat{\gamma}_k^+}}} \right) .$$

Note that $\widehat{\gamma}_k^{\pm}$ involves estimation of the spectral norm of the Hessian, which is computationally cheaper to evaluate compared to the complete Hessian matrix [26]. This is particularly relevant for deep learning, where computation of the Hessian matrix can be prohibitively expensive.

6 Conclusions and Future Work

In this work we proved data-dependent stability bounds for SGD and revisited its generalization ability. We presented novel bounds for convex and non-convex smooth loss functions, partially controlled by data-dependent quantities, while previous stability bounds for SGD were derived through the worst-case analysis. In particular, for non-convex learning, we demonstrated theoretically that generalization of SGD is heavily affected by the expected curvature around the initialization point. We demonstrated empirically that our bound is indeed tighter compared to the uniform one. In addition, our data-dependent analysis also allowed us to show optimistic bounds on the generalization error of SGD, which exhibit fast rates subject to the vanishing empirical risk of the algorithm's output.

In future work we further intend to explore our theoretical findings experimentally and evaluate the feasibility of the transfer learning based on the second-order information. Another direction lies in making our bounds adaptive. So far we have presented bounds that have data-dependent components, however the step size cannot be adjusted depending on the data, e.g. as in [35]. This was partially addressed by [23], albeit in the context of uniform stability, and we plan to extend this idea to the context of data-dependent stability.

References

[1] Z. Allen-Zhu and E. Hazan. Variance reduction for faster non-convex optimization. In *International Conference on Machine Learing (ICML)*, pages 699–707, 2016.

[2] S. Ben-David and R. Urner. Domain adaptation as learning with auxiliary information. NIPS workshop on New Directions in Transfer and Multi-Task, 2013.

[3] S. Boucheron, G. Lugosi, and P. Massart. *Concentration inequalities: A nonasymptotic theory of independence*. Oxford University Press, 2013.

[4] O. Bousquet and A. Elisseeff. Stability and Generalization. *Journal of Machine Learning Research*, 2:499–526, 2002.

[5] P. Chaudhari, A. Choromanska, S. Soatto, Y. LeCun, C. Baldassi, C. Borgs, J. Chayes, L. Sagun, and R. Zecchina. Entropy-sgd: Biasing gradient descent into wide valleys. In *International Conference on Learning Representations (ICLR)*, 2017.

[6] A. Choromanska, M. Henaff, M. Mathieu, G. B. Arous, and Y. LeCun. The loss surfaces of multilayer networks. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2015.

[7] F. Cucker and D. X. Zhou. *Learning theory: an approximation theory viewpoint*, volume 24. Cambridge University Press, 2007.

[8] L. Devroye and T. Wagner. Distribution-free performance bounds for potential function rules. *IEEE Transactions on Information Theory*, 25(5):601–604, 1979.

[9] A. Elisseeff, T. Evgeniou, and M. Pontil. Stability of randomized learning algorithms. *Journal of Machine Learning Research*, 6(Jan):55–79, 2005.

[10] T. Galanti, L. Wolf, and T. Hazan. A theoretical framework for deep transfer learning. *Information and Inference*, page iaw008, 2016.

[11] R. Ge, F. Huang, C. Jin, and Y. Yuan. Escaping from saddle points-online stochastic gradient for tensor decomposition. In *Conference on Learning Theory (COLT)*, pages 797–842, 2015.

[12] S. Ghadimi and G. Lan. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013.

[13] A. Gonen and S. Shalev-Shwartz. Fast rates for empirical risk minimization of strict saddle problems. *arXiv preprint arXiv:1701.04271*, 2017.

[14] I. Goodfellow, Y. Bengio, and A. Courville. *Deep learning*. The MIT Press, 2016.

[15] M. Hardt, B. Recht, and Y. Singer. Train faster, generalize better: Stability of stochastic gradient descent. In *International Conference on Machine Learing (ICML)*, 2016.

[16] S. Hochreiter and J. Schmidhuber. Flat minima. *Neural Computation*, 9(1):1–42, 1997.

[17] K. Kawaguchi. Deep learning without poor local minima. In *Conference on Neural Information Processing Systems (NIPS)*, pages 586–594, 2016.

[18] N. S. Keskar, D. Mudigere, J. Nocedal, M. Smelyanskiy, and P. T. P. Tang. On large-batch training for deep learning: Generalization gap and sharp minima. In *International Conference on Learning Representations (ICLR)*, 2017.

[19] T. Koren and K. Levy. Fast rates for exp-concave empirical risk minimization. In *Conference on Neural Information Processing Systems (NIPS)*, pages 1477–1485, 2015.

[20] I. Kuzborskij and F. Orabona. Stability and hypothesis transfer learning. In *International Conference on Machine Learing (ICML)*, pages 942–950, 2013.

[21] I. Kuzborskij and F. Orabona. Fast Rates by Transferring from Auxiliary Hypotheses. *Machine Learning*, pages 1–25, 2016.

[22] J. D. Lee, M. Simchowitz, M. I. Jordan, and B. Recht. Gradient descent only converges to minimizers. In *Conference on Learning Theory (COLT)*, pages 1246–1257, 2016.

[23] B. London. Generalization bounds for randomized learning with application to stochastic gradient descent. In *NIPS Workshop on Optimizing the Optimizers*, 2016.

[24] A. Maurer and M. Pontil. Empirical bernstein bounds and sample variance penalization. *arXiv preprint arXiv:0907.3740*, 2009.

[25] F. Orabona. Simultaneous model selection and optimization through parameter-free stochastic learning. In *Conference on Neural Information Processing Systems (NIPS)*, pages 1116–1124, 2014.

[26] B. A. Pearlmutter. Fast exact multiplication by the hessian. *Neural Computation*, 6(1):147–160, 1994.

[27] A. Pentina and C. H. Lampert. A pac-bayesian bound for lifelong learning. In *International Conference on Machine Learing (ICML)*, 2014.

[28] M. Perrot and A. Habrard. A theoretical analysis of metric hypothesis transfer learning. In *International Conference on Machine Learing (ICML)*, pages 1708–1717, 2015.

[29] T. Poggio, S. Voinea, and L. Rosasco. Online learning, stability, and stochastic gradient descent. *arXiv preprint arXiv:1105.4701*, 2011.

[30] S. J. Reddi, A. Hefny, S. Sra, B. Poczos, and A. Smola. Stochastic variance reduction for nonconvex optimization. In *International Conference on Machine Learing (ICML)*, pages 314–323, 2016.

[31] S. Shalev-Shwartz and S. Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014.

[32] S. Shalev-Shwartz, O. Shamir, N. Srebro, and K. Sridharan. Learnability, stability and uniform convergence. *Journal of Machine Learning Research*, 11(Oct):2635–2670, 2010.

[33] T. Tommasi, F. Orabona, and B. Caputo. Learning categories from few examples with multi model knowledge transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(5):928–941, 2014.

[34] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals. Understanding deep learning requires rethinking generalization. In *International Conference on Learning Representations (ICLR)*, 2017.

[35] P. Zhao and T. Zhang. Stochastic optimization with importance sampling for regularized loss minimization. In *International Conference on Machine Learing (ICML)*, pages 1–9, 2015.

Acknowledgments

This work was in parts funded by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement no 637076).

This work was in parts funded by the European Research Council under the European Union's Seventh Framework Programme (FP7/2007-2013)/ERC grant agreement no 308036.

A Proofs

In this section we present proofs of all the statements.

Proof of Theorem 2. Indicate by $S = \{z_i\}_{i=1}^m$ and $S' = \{z_i'\}_{i=1}^m$ independent training sets sampled i.i.d. from $\mathcal{D}$, and let $S^{(i)} = \{z_1, \ldots, z_{i-1}, z_i', z_{i+1}, \ldots, z_m\}$, such that $z_i' \overset{\text{iid}}{\sim} \mathcal{D}$. We relate expected empirical risk and expected risk by

$$\begin{aligned}
\mathop{\mathbb{E}}_S \mathop{\mathbb{E}}_A \left[\widehat{R}_S(A_S)\right] &= \mathop{\mathbb{E}}_S \mathop{\mathbb{E}}_A \left[\frac{1}{m}\sum_{i=1}^m f(A_S, z_i)\right] \\
&= \mathop{\mathbb{E}}_{S,S'} \mathop{\mathbb{E}}_A \left[\frac{1}{m}\sum_{i=1}^m f(A_{S^{(i)}}, z_i')\right] \\
&= \mathop{\mathbb{E}}_{S,S'} \mathop{\mathbb{E}}_A \left[\frac{1}{m}\sum_{i=1}^m f(A_S, z_i')\right] - \delta \\
&= \mathop{\mathbb{E}}_S \mathop{\mathbb{E}}_A [R(A_S)] - \delta \,,
\end{aligned}$$

where

$$\begin{aligned}
\delta &= \mathop{\mathbb{E}}_{S,S'} \mathop{\mathbb{E}}_A \left[\frac{1}{m}\sum_{i=1}^m \left(f(A_S, z_i') - f(A_{S^{(i)}}, z_i')\right)\right] \\
&= \frac{1}{m}\sum_{i=1}^m \mathop{\mathbb{E}}_{S,z_i'} \mathop{\mathbb{E}}_A \left[f(A_S, z_i') - f(A_{S^{(i)}}, z_i')\right] \,.
\end{aligned}$$

Renaming z_i' as z and taking sup over i we get that

$$\delta \leqslant \sup_{i \in [m]} \left\{ \mathop{\mathbb{E}}_{S,z} \mathop{\mathbb{E}}_A [f(A_S, z) - f(A_{S^{(i)}}, z)] \right\} \,.$$

This completes the proof. □

A.1 Preliminaries

We say that the SGD gradient update rule is an operator $G_t : \mathcal{H} \mapsto \mathcal{H}$, such that

$$G_t(\boldsymbol{w}) := \boldsymbol{w} - \alpha_t \nabla f(\boldsymbol{w}, z_{i_t}) \,,$$

and it is also a function of the training set S and a random index set I. Then, $\boldsymbol{w}_{t+1} = G_t(\boldsymbol{w}_t)$, throughout $t = 1, \ldots, T$. Recall the use of notation $\boldsymbol{w}_{S,t}$ to indicate the output of SGD ran on a training set S, at step t, and define

$$\delta_t(S, z) := \|\boldsymbol{w}_{S,t} - \boldsymbol{w}_{S^{(i)},t}\| \,.$$

Next, we summarize a few instrumental facts about G_t and few statements about the loss functions used in our proofs.

Definition 6 (Expansiveness)**.** *A gradient update rule is η-expansive if for all $\boldsymbol{w}, \boldsymbol{v}$,*

$$\|G_t(\boldsymbol{w}) - G_t(\boldsymbol{v})\| \leqslant \eta \|\boldsymbol{w} - \boldsymbol{v}\| .$$

The following lemma characterizes expansiveness for the gradient update rule under different assumptions on f.

Lemma 1 (Lemma 3.6 in [15])**.** *Assume that f is β-smooth. Then, we have that:*

1) G_t is $(1 + \alpha_t\beta)$-expansive,

2) If f in addition is convex, then, for any $\alpha_t \leqslant \frac{2}{\beta}$, the gradient update rule G_t is 1-expansive.

An important consequence of β-smoothness of f is self-boundedness [31], which we will use on many occasions.

Lemma 2 (Self-boundedness)**.** *For β-smooth non-negative function f we have that*

$$\|\nabla f(\boldsymbol{w}, z)\| \leqslant \sqrt{2\beta f(\boldsymbol{w}, z)} .$$

Self-boundedness in turn implies the following boundedness of a gradient update rule.

Corollary 3. *Assume that f is β-smooth and non-negative. Then,*

$$\|\boldsymbol{w} - G_t(\boldsymbol{w})\| = \alpha_t \|\nabla f(\boldsymbol{w}, z_{j_t})\| \leqslant \alpha_t \min\left\{\sqrt{2\beta f(\boldsymbol{w}, z_{j_t})}, L\right\} .$$

Proof. By Lemma 2

$$\|\alpha_t \nabla f(\boldsymbol{w}, z_{j_t})\| \leqslant \alpha_t \sqrt{2\beta f(\boldsymbol{w}, z_{j_t})} ,$$

and also by Lipschitzness of f, $\|\alpha_t \nabla f(\boldsymbol{w}, z_{j_t})\| \leqslant \alpha_t L$. □

Next we introduce a bound that relates the risk of the output at step t to the risk of the initialization point $\boldsymbol{w}_1$ through the variance of the gradient. Given an appropriate choice of step size, this bound will be crucial at stating stability bounds that depend on the risk at $\boldsymbol{w}_1$. The proof idea is similar to the one of [12]. In particular, it does not require convexity of the loss function.

Lemma 3. *Suppose SGD is ran with step sizes $\alpha_1, \dots, \alpha_{t-1} \leqslant \frac{1}{\beta}$ w.r.t. the β-smooth loss f. Then we have that*

$$\sum_{k=1}^{t-1} \left(\alpha_k - \frac{\alpha_k^2 \beta}{2}\right) \mathop{\mathbb{E}}_S\left[\|\nabla R(\boldsymbol{w}_k)\|^2\right] \leqslant R(\boldsymbol{w}_1) - R(\boldsymbol{w}_t) + \frac{\beta}{2} \sum_{k=1}^{t-1} \alpha_k^2 \mathop{\mathbb{E}}_S\left[\|\nabla f(\boldsymbol{w}_k, z_{j_k}) - \nabla R(\boldsymbol{w}_k)\|^2\right] . \tag{4}$$

Proof. For brevity denote $f_k(\boldsymbol{w}) \equiv f(\boldsymbol{w}, z_{j_k})$. By β-smoothness of R and recalling that the SGD update rule $\boldsymbol{w}_{k+1} = \boldsymbol{w}_k - \alpha_k \nabla f_k(\boldsymbol{w}_k)$, we have

$$\begin{aligned}
R(\boldsymbol{w}_{k+1}) - R(\boldsymbol{w}_k) &\leqslant \nabla R(\boldsymbol{w}_k)^\top (\boldsymbol{w}_{k+1} - \boldsymbol{w}_k) + \frac{\beta}{2}\|\boldsymbol{w}_{k+1} - \boldsymbol{w}_k\|^2 \\
&= -\alpha_k \nabla R(\boldsymbol{w}_k)^\top \nabla f_k(\boldsymbol{w}_k) + \frac{\beta\alpha_k^2}{2}\|\nabla f_k(\boldsymbol{w}_k)\|^2 \\
&= -\alpha_k \nabla R(\boldsymbol{w}_k)^\top \nabla f_k(\boldsymbol{w}_k) + \frac{\beta\alpha_k^2}{2}\|\nabla f_k(\boldsymbol{w}_k) - \nabla R(\boldsymbol{w}_k) + \nabla R(\boldsymbol{w}_k)\|^2 \\
&= -\alpha_k \nabla R(\boldsymbol{w}_k)^\top \nabla f_k(\boldsymbol{w}_k) \\
&+ \frac{\beta\alpha_k^2}{2}\Big(\|\nabla f_k(\boldsymbol{w}_k) - \nabla R(\boldsymbol{w}_k)\|^2 + \|\nabla R(\boldsymbol{w}_k)\|^2 \\
&- 2\left(\nabla f_k(\boldsymbol{w}_k) - \nabla R(\boldsymbol{w}_k)\right)^\top \nabla R(\boldsymbol{w}_k)\Big) \\
&= -\left(\alpha_k + \alpha_k^2\beta\right) \nabla R(\boldsymbol{w}_k)^\top \nabla f_k(\boldsymbol{w}_k) \\
&+ \frac{3\alpha^2\beta}{2}\|\nabla R(\boldsymbol{w}_k)\|^2 + \frac{\beta\alpha_k^2}{2}\|\nabla f_k(\boldsymbol{w}_k) - \nabla R(\boldsymbol{w}_k)\|^2 .
\end{aligned}$$

Taking expectation w.r.t. S on both sides, recalling that $\mathbb{E}_{z_k}[\nabla f_k(\boldsymbol{w}_k)] = \nabla R(\boldsymbol{w}_k)$ and rearranging terms we get

$$\left(\alpha_k - \frac{\alpha^2\beta}{2}\right) \mathbb{E}\left[\|\nabla R(\boldsymbol{w}_k)\|^2\right] \leqslant R(\boldsymbol{w}_k) - R(\boldsymbol{w}_{k+1}) + \frac{\beta\alpha_k^2}{2} \mathbb{E}\left[\|\nabla f_k(\boldsymbol{w}_k) - \nabla R(\boldsymbol{w}_k)\|^2\right] \ ,$$

and summing above over $k = 1, \dots, t-1$ we get the statement. □

Lemma 4. *Suppose SGD is ran with step sizes $\alpha_1, \dots, \alpha_{t-1} \leqslant \frac{1}{\beta}$ on the β-smooth loss f. Assume that the variance of stochastic gradients obeys*

$$\mathop{\mathbb{E}}_{S,z}\left[\|\nabla f(\boldsymbol{w}_{S,k}, z) - \nabla R(\boldsymbol{w}_{S,k})\|^2\right] \leqslant \sigma^2 \quad \forall k \in [T] \ .$$

Then we have that

$$\mathop{\mathbb{E}}_{S}\left[\sum_{k=1}^{t-1} \alpha_k \|\nabla f(\boldsymbol{w}_{S,k}, z_k)\|\right] \leqslant 2\sqrt{\left(\sum_{k=1}^{t-1}\alpha_k\right)\left(R(\boldsymbol{w}_1) - \inf_{\boldsymbol{w}\in\mathcal{H}} R(\boldsymbol{w}) + \frac{\beta\sigma^2}{2}\sum_{k=1}^{t-1}\alpha_k^2\right)} + \sigma\sum_{k=1}^{t-1}\alpha_k \ .$$

Proof. First we perform the decomposition,

$$\begin{aligned}\mathop{\mathbb{E}}_{S}\left[\sum_{k=1}^{t-1} \alpha_k \|\nabla f(\boldsymbol{w}_{S,k}, z_k)\|\right] &= \sum_{k=1}^{t-1}\alpha_k \mathop{\mathbb{E}}_{S}[\|\nabla R(\boldsymbol{w}_{S,k})\|] + \sum_{k=1}^{t-1}\alpha_k \mathop{\mathbb{E}}_{S}[\|\nabla f(\boldsymbol{w}_{S,k}, z_k) - \nabla R(\boldsymbol{w}_{S,k})\|] \\ &\leqslant \sum_{k=1}^{t-1}\alpha_k \mathop{\mathbb{E}}_{S}[\|\nabla R(\boldsymbol{w}_{S,k})\|] + \sigma\sum_{k=1}^{t-1}\alpha_k \ . \qquad (5)\end{aligned}$$

Introduce

$$Q_t := \sum_{k=1}^{t-1}\left(\alpha_k - \frac{\alpha_k^2\beta}{2}\right) \ .$$

Now we invoke the stationary-point argument to bound the first term above as

$$\begin{aligned}\sum_{k=1}^{t-1}\alpha_k \mathop{\mathbb{E}}_{S}\left[\sqrt{\|\nabla R(\boldsymbol{w}_k)\|^2}\right] &\leqslant \sum_{k=1}^{t-1}\frac{\left(1-\frac{\alpha_k\beta}{2}\right)}{\left(1-\frac{\alpha_k\beta}{2}\right)}\cdot\alpha_k\sqrt{\mathop{\mathbb{E}}_{S}\left[\|\nabla R(\boldsymbol{w}_k)\|^2\right]} && \text{(By Jensen's inequality)} \\ &\leqslant 2\sum_{k=1}^{t-1}\left(\alpha_k - \frac{\alpha_k^2\beta}{2}\right)\sqrt{\mathop{\mathbb{E}}_{S}\left[\|\nabla R(\boldsymbol{w}_k)\|^2\right]} && \text{(Assuming that } \alpha_k \leqslant \tfrac{1}{\beta}) \\ &= \frac{2Q_t}{Q_t}\sum_{k=1}^{t-1}\left(\alpha_k - \frac{\alpha_k^2\beta}{2}\right)\sqrt{\mathop{\mathbb{E}}_{S}\left[\|\nabla R(\boldsymbol{w}_k)\|^2\right]} && (6) \\ &\leqslant 2\sqrt{Q_t}\sqrt{\sum_{k=1}^{t-1}\left(\alpha_k - \frac{\alpha_k^2\beta}{2}\right)\mathop{\mathbb{E}}_{S}\left[\|\nabla R(\boldsymbol{w}_k)\|^2\right]} && \text{(By Jensen's inequality)} \\ &\leqslant 2\sqrt{Q_t}\sqrt{R(\boldsymbol{w}_1) - R(\boldsymbol{w}_t) + \frac{\beta\sigma^2}{2}\sum_{k=1}^{t-1}\alpha_k^2} \ . && \text{(By Lemma 3)}\end{aligned}$$

Combining this with (5) gives

$$\mathop{\mathbb{E}}_{S}\left[\sum_{k=1}^{t-1} \alpha_k \|\nabla f(\boldsymbol{w}_{S,k}, z_k)\|\right] \leqslant 2\sqrt{\left(\sum_{k=1}^{t-1}\alpha_k\right)\left(R(\boldsymbol{w}_1) - \inf_{\boldsymbol{w}\in\mathcal{H}} R(\boldsymbol{w}) + \frac{\beta\sigma^2}{2}\sum_{k=1}^{t-1}\alpha_k^2\right)} + \sigma\sum_{k=1}^{t-1}\alpha_k \ , \qquad (7)$$

which completes the proof. □

The following lemma is similar to Lemma 3.11 of [15], and is instrumental in bounding the stability of SGD. However, we make an adjustment and state it in expectation over the data. Note that it does not require convexity of the loss function.

Lemma 5. *Assume that the loss function $f(\cdot, z) \in [0,1]$ is L-Lipschitz for all z. Then, for every $t_0 \in \{0, 1, 2, \dots m\}$ we have that,*

$$\mathop{\mathbb{E}}_{S,z} \mathop{\mathbb{E}}_{A} \left[f(\boldsymbol{w}_{S,T}, z) - f(\boldsymbol{w}_{S^{(i)},T}, z) \right] \tag{8}$$

$$\leqslant L \mathop{\mathbb{E}}_{S,z} \left[\mathop{\mathbb{E}}_{A} [\delta_T(S,z) \mid \delta_{t_0}(S,z) = 0] \right] + \mathop{\mathbb{E}}_{S,A} [R(A_S)] \frac{t_0}{m} \,. \tag{9}$$

Proof. We proceed with elementary decomposition, Lipschitzness of f, and using the fact that f is non-negative to have that

$$\begin{aligned} f(\boldsymbol{w}_{S,T}, z) - f(\boldsymbol{w}_{S^{(i)},T}, z) &= \big(f(\boldsymbol{w}_{S,T}, z) - f(\boldsymbol{w}_{S^{(i)},T}, z)\big) \mathbb{I}\{\delta_{t_0}(S,z) = 0\} \qquad (10) \\ &\quad + \big(f(\boldsymbol{w}_{S,T}, z) - f(\boldsymbol{w}_{S^{(i)},T}, z)\big) \mathbb{I}\{\delta_{t_0}(S,z) \neq 0\} \\ &\leqslant L\delta_T(S,z)\mathbb{I}\{\delta_{t_0}(S,z) = 0\} + f(\boldsymbol{w}_{S,T}, z)\mathbb{I}\{\delta_{t_0}(S,z) \neq 0\} \,. \qquad (11) \end{aligned}$$

Taking expectation w.r.t. algorithm randomization, we get that

$$\begin{aligned} \mathop{\mathbb{E}}_{A} \left[f(\boldsymbol{w}_{S,T}, z) - f(\boldsymbol{w}_{S^{(i)},T}, z) \right] &\leqslant L \mathop{\mathbb{E}}_{A} [\delta_T(S,z)\mathbb{I}\{\delta_{t_0}(S,z) = 0\}] \qquad (12) \\ &\quad + \mathop{\mathbb{E}}_{A} [f(\boldsymbol{w}_{S,T}, z)\mathbb{I}\{\delta_{t_0}(S,z) \neq 0\}] \,. \qquad (13) \end{aligned}$$

Recall that $i \in [m]$ is the index where S and $S^{(i)}$ differ, and introduce a random variable τ_A taking on the index of the first time step where SGD uses the example z_i or a replacement z. Note also that τ_A does not depend on the data. When $\tau_A > t_0$, then it must be that $\delta_{t_0}(S,z) = 0$, because updates on both S and $S^{(i)}$ are identical until t_0. A consequence of this is that $\mathbb{I}\{\delta_{t_0}(S,z) \neq 0\} \leqslant \mathbb{I}\{\tau_A \leqslant t_0\}$. Thus the rightmost term in (13) is bounded as

$$\mathop{\mathbb{E}}_{A} [f(\boldsymbol{w}_{S,T}, z)\mathbb{I}\{\delta_{t_0}(S,z) \neq 0\}] \leqslant \mathop{\mathbb{E}}_{A} [f(\boldsymbol{w}_{S,T}, z)\mathbb{I}\{\tau_A \leqslant t_0\}] \,.$$

Now, focus on the r.h.s. above. Recall that we assume randomization by sampling from the uniform distribution over $[m]$ without replacement, and denote a realization by $\{j_i\}_{i=1}^m$. Then, we can always express our randomization as permutation function $\pi_A(S) = \{z_{j_i}\}_{i=1}^m$. In addition, introduce an algorithm GD : $\mathcal{Z}^m \mapsto \mathcal{H}$, which is identical to A, except that it passes over the training set S sequentially without randomization. That said, we have that

$$\mathop{\mathbb{E}}_{A} [f(\boldsymbol{w}_{S,T}, z)\mathbb{I}\{\tau_A \leqslant t_0\}] = \mathop{\mathbb{E}}_{A} \left[f(\mathrm{GD}_{\pi_A(S)}, z)\mathbb{I}\{\tau_A \leqslant t_0\} \right] ,$$

and taking expectation over the data,

$$\mathop{\mathbb{E}}_{S,z} \left[\mathop{\mathbb{E}}_{A} [f(\boldsymbol{w}_{S,T}, z)\mathbb{I}\{\tau_A \leqslant t_0\}] \right] = \mathop{\mathbb{E}}_{A} \left[\mathop{\mathbb{E}}_{S,z} \left[f(\mathrm{GD}_{\pi_A(S)}, z) \right] \mathbb{I}\{\tau_A \leqslant t_0\} \right] .$$

Now observe that for any realization of A, $\mathbb{E}_{S,z}\left[f(\mathrm{GD}_{\pi_A(S)}, z)\right] = \mathbb{E}_A \mathbb{E}_{S,z}[f(A_S, z)]$ because expectation w.r.t. S and z does not change under our randomization [1]. Thus, we have that

$$\mathop{\mathbb{E}}_{A} \left[\mathop{\mathbb{E}}_{S,z} \left[f(\mathrm{GD}_{\pi_A(S)}, z) \right] \mathbb{I}\{\tau_A \leqslant t_0\} \right] = \mathop{\mathbb{E}}_{S,A} [R(A_S)] \, \mathbb{P}(\tau_A \leqslant t_0) \,.$$

[1] Strictly speaking we could omit $\mathbb{E}_A[\cdot]$ and consider *any* randomization by reshuffling, but we keep expectation for the sake of clarity.

Now assuming that τ_A is uniformly distributed over $[m]$ we have that

$$\mathbb{P}(\tau_A \leqslant t_0) = \frac{t_0}{m} .$$

Putting this together with (10) and (11), we finally get that

$$\begin{aligned}\mathop{\mathbb{E}}_{S,z}\mathop{\mathbb{E}}_{A}\left[f(\boldsymbol{w}_{S,T}, z) - f(\boldsymbol{w}_{S^{(i)},T}, z)\right] &\leqslant L \mathop{\mathbb{E}}_{S,z}\left[\mathop{\mathbb{E}}_{A}\left[\delta_T(S,z)\mathbb{I}\left\{\delta_{t_0}(S,z) = 0\right\}\right]\right] + \mathop{\mathbb{E}}_{S,A}\left[R(A_S)\right]\frac{t_0}{m} \\ &\leqslant L \mathop{\mathbb{E}}_{S,z}\left[\mathop{\mathbb{E}}_{A}\left[\delta_T(S,z) \mid \delta_{t_0}(S,z) = 0\right]\right] + \mathop{\mathbb{E}}_{S,A}\left[R(A_S)\right]\frac{t_0}{m} .\end{aligned}$$

This completes the proof. □

We spend a moment to highlight the role of conditional expectation in (9). Observe that we could naively bound (8) by the Lipschitzness of f, but Lemma 5 follows a more careful argument. First note that t_0 is a free parameter. The expected distance in (9) between SGD outputs $\boldsymbol{w}_{S,t}$ and $\boldsymbol{w}_{S^{(i)},t}$ is conditioned on the fact that at step t_0 outputs of SGD are still the same. This means that the perturbed point is encountered after t_0. Then, the conditional expectation should be a decreasing function of t_0: the later the perturbation occurs, the smaller deviation between $\boldsymbol{w}_{S,t}$ and $\boldsymbol{w}_{S^{(i)},t}$ we should expect. Later we use this fact to minimize the bound (9) over t_0.

A.2 Convex Losses

In this section we prove on-average stability for loss functions that are non-negative, β-smooth, and convex.

Theorem 5. *Assume that f is convex, and that SGD's is ran with step sizes $\{\alpha_t\}_{t=1}^T$. Then, for every $t_0 \in \{0,1,2,\dots m\}$, SGD is $\epsilon(\mathcal{D}, \boldsymbol{w}_1)$-on-average stable with*

$$\epsilon(\mathcal{D}, \boldsymbol{w}_1) \leqslant \frac{2}{m}\sum_{t=t_0+1}^{T} \alpha_t \mathop{\mathbb{E}}_{S,z}\left[\|\nabla f(\boldsymbol{w}_t, z_{j_t})\|\right] + \mathop{\mathbb{E}}_{S,A}\left[R(A_S)\right]\frac{t_0}{m} .$$

Proof. For brevity denote $\Delta_t(S,z) := \mathbb{E}_A\left[\delta_t(S,z) \mid \delta_{t_0}(S,z) = 0\right]$. We start by applying Lemma 5:

$$\mathop{\mathbb{E}}_{S,z}\mathop{\mathbb{E}}_{A}\left[f(\boldsymbol{w}_{S,T}, z) - f(\boldsymbol{w}_{S^{(i)},T}, z)\right] \leqslant L \mathop{\mathbb{E}}_{S,z}\left[\Delta_T(S,z)\right] + \mathop{\mathbb{E}}_{S,A}\left[R(A_S)\right]\frac{t_0}{m} . \tag{14}$$

Our goal is to bound the first term on the r.h.s. as a decreasing function of t_0, so that eventually we can minimize the bound w.r.t. t_0. At this point we focus on the first term, and the proof partially follows the outline of the proof of Theorem 3.7 in [15]. The strategy will be to establish the bound on $\Delta_T(S,z)$ by using a recursive argument. In fact we will state the bound on $\Delta_{t+1}(S,z)$ in terms of $\Delta_t(S,z)$ and then unravel the recursion. Finally, we will take expectation w.r.t. the data after we obtain the bound by recursion.

To do so, we distinguish two cases: 1) SGD encounters a perturbed point at step t, that is $t = i$, and 2) the current point is the same in S and $S^{(i)}$, so $t \neq i$. For the first case, we will use data-dependent boundedness of the gradient update rule, Corollary 3, that is

$$\|G_t(\boldsymbol{w}_{S,t}) - G_t(\boldsymbol{w}_{S^{(i)},t})\| \leqslant \delta_t(S,z) + 2\alpha_t\|\nabla f(\boldsymbol{w}_{S,t}, z_{j_t})\| .$$

To handle the second case, we will use the expansiveness of the gradient update rule, Lemma 1, which states that for convex loss functions, the gradient update rule is 1-expansive, so $\delta_{t+1}(S,z) \leqslant \delta_t(S,z)$. Considering both cases of example selection, and noting that SGD encounters the perturbation w.p. $\frac{1}{m}$, we write $\mathbb{E}_A$ for a step t as

$$\begin{aligned}\Delta_{t+1}(S,z) &\leqslant \left(1 - \frac{1}{m}\right)\Delta_t(S,z) + \frac{1}{m}\left(\Delta_t(S,z) + 2\alpha_t\|\nabla f(\boldsymbol{w}_{S,t}, z_{j_t})\|\right) \\ &= \Delta_t(S,z) + \frac{2\alpha_t\|\nabla f(\boldsymbol{w}_{S,t}, z_{j_t})\|}{m} .\end{aligned}$$

Unraveling the recursion from T to t_0 and plugging the above into (14) yields

$$\mathop{\mathbb{E}}_{A}\mathop{\mathbb{E}}_{S,z}[\delta_T(S,z)] \leqslant \frac{2}{m}\sum_{t=t_0+1}^{T}\alpha_t \mathop{\mathbb{E}}_{S,z}\left[\|\nabla f(\boldsymbol{w}_t, z_{j_t})\|\right] + \mathop{\mathbb{E}}_{S,A}[R(A_S)]\,\frac{t_0}{m}\,.$$

This completes the proof. □

Next statement is a simple consequence of Theorem 5 and Lemma 4.

Proof of Theorem 3. Consider Theorem 5 and set $t_0 = 0$.

$$\epsilon(\mathcal{D}, \boldsymbol{w}_1) \leqslant \frac{2}{m}\sum_{t=1}^{T}\alpha_t \mathop{\mathbb{E}}_{S,z}\left[\|\nabla f(\boldsymbol{w}_{S,t}, z_{j_t})\|\right]\,. \tag{15}$$

Bounding the sum using Lemma 4 recalling that $\alpha_t = c/\sqrt{t}$, we get

$$\begin{aligned}\mathop{\mathbb{E}}_{S}\left[\sum_{t=1}^{T}\alpha_t\|\nabla f(\boldsymbol{w}_t, z_{j_t})\|\right] &\leqslant 2\sqrt{\left(\sum_{t=1}^{T}\alpha_t\right)\left(R(\boldsymbol{w}_1) - R^\star + \frac{\beta\sigma^2}{2}\sum_{t=1}^{T}\alpha_t^2\right)} + \sigma\sum_{t=1}^{T}\alpha_t \\ &\leqslant 2\sqrt{2c}\cdot\sqrt[4]{T}\cdot\sqrt{R(\boldsymbol{w}_1) - R^\star} + 2c\sigma\left(\sqrt[4]{T}\sqrt{\frac{\beta}{2}} + \sqrt{T}\right)\,.\end{aligned}$$

Combining above with (15) completes the proof. □

A.3 Non-convex Losses

Our proof of a stability bound for non-convex loss functions, Theorem 4 (in the submission file), follows a general outline of [15, Theorem 3.8]. Namely, the outputs of SGD run on a training set S and its perturbed version $S^{(i)}$ will not differ too much, because by the time a perturbation is encountered, the step size has already decayed enough. So, on the one hand, stabilization is enforced by the diminishing the step size, and on the other hand, by how much updates expand the distance between the gradients after the perturbation. Since [15] work with uniform stability, they capture the expansiveness of post-perturbation update by the Lipschitzness of the gradient. In combination with a recursive argument, their bound has exponential dependency on the Lipschitz constant of the gradient. We argue that the Lipschitz continuity of the gradient can be too pessimistic in general. Instead, we rely on a local data-driven argument: considering that we initialize SGD at point $\boldsymbol{w}_1$, how much do updates expand the gradient under the distribution of interest? The following crucial lemma characterizes such behavior in terms of the curvature at $\boldsymbol{w}_1$.

Lemma 6. *Assume that the loss function $f(\cdot, z)$ is β-smooth and that its Hessian is ρ-Lipschitz. Then,*

$$\left\|G_t(\boldsymbol{w}_{S,t}) - G_t(\boldsymbol{w}_{S^{(i)},t})\right\| \leqslant \left(1 + \alpha_t\xi_t(S,z)\right)\delta_t(S,z) \tag{16}$$

where

$$\xi_t(S,z) := \left\|\nabla^2 f(\boldsymbol{w}_1, z_t)\right\|_2 + \frac{\rho}{2}\left\|\sum_{k=1}^{t-1}\alpha_k\nabla f(\boldsymbol{w}_{S,k}, z_k)\right\| + \frac{\rho}{2}\left\|\sum_{k=1}^{t-1}\alpha_k\nabla f(\boldsymbol{w}_{S^{(i)},k}, z_{k'})\right\|\,.$$

Furthermore, for any $t \in [T]$,

$$\begin{aligned}\mathop{\mathbb{E}}_{S,z}[\xi_t(S,z)] \leqslant \mathop{\mathbb{E}}_{S,z}&\left[\left\|\nabla^2 f(\boldsymbol{w}_1, z_t)\right\|_2\right] \\ &+ 2\rho\sqrt{\left(R(\boldsymbol{w}_1) - R^\star\right)c(1 + \ln(T))} \\ &+ \rho\sigma\left(\sqrt{2c\beta} + c(1 + \ln(T))\right)\,.\end{aligned}$$

Proof. Recall that the randomness of the algorithm is realized through sampling without replacement from the uniform distribution over $[m]$. Apart from that we will not be concerned with the randomness of the algorithm, and given the set of random variables $\{j_i\}_{i=1}^m$, for brevity we will use indexing notation $z_1, z_2, \dots, z_m$ to indicate $z_{j_1}, z_{j_2}, \dots, z_{j_m}$. Next, let $S^{(i)} = \{z_i'\}_{i=1}^m$, and introduce a shorthand notation $f_k(\boldsymbol{w}) = f(\boldsymbol{w}, z_k)$ and $f_{k'}(\boldsymbol{w}) = f(\boldsymbol{w}, z_k')$. We start by applying triangle inequality to get

$$\left\|G_t(\boldsymbol{w}_{S,t}) - G_t(\boldsymbol{w}_{S^{(i)},t})\right\| \leqslant \|\boldsymbol{w}_{S,t} - \boldsymbol{w}_{S^{(i)},t}\| + \alpha_t \left\|\nabla f_t(\boldsymbol{w}_{S,t}) - \nabla f_t(\boldsymbol{w}_{S^{(i)},t})\right\| \ .$$

In the following we will focus on the second term of r.h.s. above. Given SGD outputs $\boldsymbol{w}_{S,t}$ and $\boldsymbol{w}_{S^{(i)},t}$ with $t > i$, our goal here is to establish how much do gradients grow apart with every new update. This behavior can be characterized assuming that gradient is Lipschitz continuous, however, we conduct a local analysis. Specifically, we observe how much do updates expand gradients, given that we start at some point $\boldsymbol{w}_1$ under the data-generating distribution. So, instead of the Lipschitz constant, expansiveness rather depends on the curvature around $\boldsymbol{w}_1$. On the other hand, we are dealing with outputs at an arbitrary time step t, and therefore we first have to relate them to the initialization point $\boldsymbol{w}_1$. We do so by using the gradient update rule and telescopic sums, and conclude that this relationship is controlled by the sum of gradient norms along the update path. We further establish that this sum is controlled by the risk of $\boldsymbol{w}_1$ up to the noise of stochastic gradients, through stationary-point result of Lemma 4. Thus, the proof consists of two parts: 1) Decomposition into curvature and gradients along the update path, and 2) bounding those gradients.

1) Decomposition. Introduce $\boldsymbol{\delta}_t := \boldsymbol{w}_{S^{(i)},t} - \boldsymbol{w}_{S,t}$. By Taylor theorem we get that

$$\nabla f_t(\boldsymbol{w}_{S,t}) - \nabla f_t(\boldsymbol{w}_{S^{(i)},t}) = \nabla^2 f_t(\boldsymbol{w}_1)\boldsymbol{\delta}_t + \int_0^1 \Big(\nabla^2 f_t(\boldsymbol{w}_{S,t} + \tau\boldsymbol{\delta}_t) - \nabla^2 f_t(\boldsymbol{w}_1)\Big) \,\mathrm{d}\tau \boldsymbol{\delta}_t \ .$$

Taking norm on both sides, applying triangle inequality, Cauchy-Schwartz inequality, and assuming that Hessians are ρ-Lipschitz we obtain

$$\|\nabla f_t(\boldsymbol{w}_{S,t}) - \nabla f_t(\boldsymbol{w}_{S^{(i)},t})\| \leqslant \rho \int_0^1 \|\boldsymbol{w}_{S,t} - \boldsymbol{w}_1 + \tau\boldsymbol{\delta}_t\| \,\mathrm{d}\tau \|\boldsymbol{\delta}_t\| + \left\|\nabla^2 f_t(\boldsymbol{w}_1)\right\| \|\boldsymbol{\delta}_t\| \ . \tag{17}$$

2) Bounding gradients. Using telescoping sums and SGD update rule we get that

$$\begin{aligned}
\boldsymbol{w}_{S,t} - \boldsymbol{w}_1 + \tau\boldsymbol{\delta}_t &= \boldsymbol{w}_{S,t} - \boldsymbol{w}_1 + \tau\left(\boldsymbol{w}_{S^{(i)},t} - \boldsymbol{w}_1 + \boldsymbol{w}_1 - \boldsymbol{w}_{S,t}\right) \\
&= \sum_{k=1}^{t-1} (\boldsymbol{w}_{S,k+1} - \boldsymbol{w}_{S,k}) \\
&+ \tau \sum_{k=1}^{t-1} \left(\boldsymbol{w}_{S^{(i)},k+1} - \boldsymbol{w}_{S^{(i)},k}\right) \\
&- \tau \sum_{k=1}^{t-1} (\boldsymbol{w}_{S,k+1} - \boldsymbol{w}_{S,k}) \\
&= (\tau - 1) \sum_{k=1}^{t-1} \alpha_k \nabla f_k(\boldsymbol{w}_{S,k}) - \tau \sum_{k=1}^{t-1} \alpha_k \nabla f_{k'}(\boldsymbol{w}_{S^{(i)},k}) \ .
\end{aligned}$$

Plugging above into the integral of (17) we have

$$\int_0^1 \left\| \sum_{k=1}^{t-1} \alpha_k \left((\tau - 1) \nabla f_k(\boldsymbol{w}_{S,k}) - \tau \nabla f_{k'}(\boldsymbol{w}_{S^{(i)},k}) \right) \right\| \mathrm{d}\tau$$
$$\leqslant \frac{1}{2} \left\| \sum_{k=1}^{t-1} \alpha_k \nabla f_k(\boldsymbol{w}_{S,k}) \right\| + \frac{1}{2} \left\| \sum_{k=1}^{t-1} \alpha_k \nabla f_{k'}(\boldsymbol{w}_{S^{(i)},k}) \right\|$$
$$\leqslant \frac{1}{2} \sum_{k=1}^{t-1} \alpha_k \| \nabla f_k(\boldsymbol{w}_{S,k}) \| + \frac{1}{2} \sum_{k=1}^{t-1} \alpha_k \| \nabla f_{k'}(\boldsymbol{w}_{S^{(i)},k}) \| \ .$$

Plugging this result back into (17) completes the proof of the first statement. The second statement comes from Lemma 4 with $\alpha_t = c/t$. □

Next, we need the following statement to prove our stability bound.

Proposition 2 (Bernstein-type inequality)**.** *Let Z be a zero-mean real-valued r.v., such that $|Z| \leqslant b$ and $\mathbb{E}[Z^2] \leqslant \sigma^2$. Then for all $|c| \leqslant \frac{1}{2b}$, we have that $\mathbb{E}\left[e^{cZ}\right] \leqslant e^{c^2\sigma^2}$.*

Proof. Stated inequality is a consequence of a Bernstein-type inequality for moment generating functions, Theorem 2.10 in [3]. Observe that zero-centered r.v. Z bounded by b satisfies Bernstein's condition, that is

$$|\,\mathbb{E}[(Z - \mathbb{E}[Z])^q]| \leqslant \frac{q!}{2}\sigma^2 b^{k-2} \qquad \text{for all integers } q \geqslant 3 \ .$$

This in turn satisfies condition for Bernstein-type inequality stating that

$$\mathbb{E}\left[\exp\left(c(Z - \mathbb{E}[Z])\right)\right] \leqslant \exp\left(\frac{c^2\sigma^2/2}{1 - b|c|}\right) \ .$$

Choosing $|c| \leqslant \frac{1}{2b}$ verifies the statement. □

Now we are ready to prove Theorem 4, which bounds the $\epsilon(\mathcal{D}, \boldsymbol{w}_1)$-on-average stability of SGD.

Proof of Theorem 4. For brevity denote

$$r := \mathop{\mathbb{E}}_{S,A}\left[R(A_S)\right]$$

and

$$\Delta_t(S, z) := \mathop{\mathbb{E}}_{A}\left[\delta_t(S, z) \mid \delta_{t_0}(S, z) = 0\right] \ .$$

By Lemma 5, for all $t_0 \in [m]$,

$$\mathop{\mathbb{E}}_{S,z} \mathop{\mathbb{E}}_{A}\left[f(\boldsymbol{w}_{S,T}, z) - f(\boldsymbol{w}_{S^{(i)},T}, z)\right] \leqslant L \mathop{\mathbb{E}}_{S,z}\left[\Delta_T(S, z)\right] + r\frac{t_0}{m} \ . \tag{18}$$

Most of the proof is dedicated to bounding the first term in (18). We deal with this similarly as in [15]. Specifically, we state the bound on $\Delta_T(S, z)$ by using a recursion. In our case, however, we also have an expectation w.r.t. the data, and to avoid complications with dependencies, we first unroll the recursion for the random quantities, and only then take the expectation. At this point the proof crucially relies on the product of exponentials arising from the recursion, and all relevant random quantities end up inside of them. We alleviate this by Proposition 2. Finally, we conclude by minimizing (18) w.r.t. t_0. Thus we have three steps: 1) recursion, 2) bounding $\mathbb{E}[\exp(\cdots)]$, and 3) tuning of t_0.

1) Recursion. We begin by stating the bound on $\Delta_T(S, z)$ by recursion. Thus we will first state the bound on $\Delta_{t+1}(S, z)$ in terms of $\Delta_t(S, z)$, and other relevant quantities and then unravel the recursion. As in the convex case, we distinguish two cases: 1) SGD encounters the perturbed point at step t, that is $t = i$, and 2) the current point is the same in S and $S^{(i)}$, so $t \neq i$. For the first case, we will use worst-case boundedness of G_t, Corollary 3, that is, $\|G_t(\boldsymbol{w}_{S,t}) - G_t(\boldsymbol{w}_{S^{(i)},t})\| \leqslant \delta_t(S, z) + 2\alpha_t L$. To handle the second case we will use Lemma 6, namely,

$$\left\|G_t(\boldsymbol{w}_{S,t}) - G_t(\boldsymbol{w}_{S^{(i)},t})\right\| \leqslant (1 + \alpha_t \xi_t(S, z))\, \delta_t(S, z)\ .$$

In addition, as a safety measure we will also take into account that the gradient update rule is at most $(1 + \alpha_t \beta)$-expansive by Lemma 1. So we will work with the function $\psi_t(S, z) := \min\{\xi_t(S, z), \beta\}$ instead of $\xi_t(S, z)$. and decompose the expectation w.r.t. A for a step t. Noting that SGD encounters the perturbed example with probability $\frac{1}{m}$,

$$\begin{aligned}
\Delta_{t+1}(S, z) &\leqslant \left(1 - \frac{1}{m}\right)(1 + \alpha_t \psi_t(S, z))\, \Delta_t(S, z) + \frac{1}{m}\left(2\alpha_t L + \Delta_t(S, z)\right) \\
&= \left(1 + \left(1 - \frac{1}{m}\right)\alpha_t \psi_t(S, z)\right)\Delta_t(S, z) + \frac{2\alpha_t L}{m} \\
&\leqslant \exp\left(\alpha_t \psi_t(S, z)\right)\Delta_t(S, z) + \frac{2\alpha_t L}{m}\ ,
\end{aligned} \tag{19}$$

where the last inequality follows from $1 + x \leqslant \exp(x)$. This inequality is not overly loose for $x \in [0, 1]$, and, in our case it becomes instrumental in handling the recursion.

Now, observe that relation $x_{t+1} \leqslant a_t x_t + b_t$ with $x_{t_0} = 0$ unwinds from T to t_0 as $x_T \leqslant \sum_{t=t_0+1}^{T} b_t \prod_{k=t+1}^{T} a_k$. Consequently, having $\Delta_{t_0}(S, z) = 0$, we unwind (19) to get

$$\begin{aligned}
\Delta_T(S, z) &\leqslant \sum_{t=t_0+1}^{T}\left(\prod_{k=t+1}^{T}\exp\left(\frac{c\psi_k(S, z)}{k}\right)\right)\frac{2cL}{mt} \\
&= \sum_{t=t_0+1}^{T}\exp\left(c\sum_{k=t+1}^{T}\frac{\psi_k(S, z)}{k}\right)\frac{2cL}{mt}\ .
\end{aligned} \tag{20}$$

2) Bounding $\mathbb{E}[\exp(\cdots)]$. We take expectation w.r.t. S and z on both sides and focus on the expectation of the exponential in (20). First, introduce $\mu_k := \mathbb{E}_{S,z}[\psi_k(S, z)]$, and proceed as

$$\mathop{\mathbb{E}}_{S,z}\left[\exp\left(c\sum_{k=t+1}^{T}\frac{\psi_k(S, z)}{k}\right)\right] = \mathop{\mathbb{E}}_{S,z}\left[\exp\left(c\sum_{k=t+1}^{T}\frac{\psi_k(S, z) - \mu_k}{k}\right)\right]\exp\left(c\sum_{k=t+1}^{T}\frac{\mu_k}{k}\right)\ . \tag{21}$$

Observe that zero-mean version of $\psi_k(S, z)$ is bounded as

$$\sum_{k=t+1}^{T}\frac{|\psi_k(S, z) - \mu_k|}{k} \leqslant 2\beta\ln(T)\ ,$$

and assume the setting of c as $c \leqslant \frac{1}{2(2\beta \ln(T))^2}$. By Proposition 2, we have

$$\begin{aligned}
\mathbb{E}\left[\exp\left(c\sum_{k=t+1}^{T}\frac{\psi_k(S,z)-\mu_k}{k}\right)\right] &\leqslant \exp\left(c^2\,\mathbb{E}\left[\left(\sum_{k=t+1}^{T}\frac{\psi_k(S,z)-\mu_k}{k}\right)^2\right]\right)\\
&= \exp\left(\frac{c}{2}\,\mathbb{E}\left[\left(\frac{1}{2\beta\ln(T)}\sum_{k=t+1}^{T}\frac{\psi_k(S,z)-\mu_k}{k}\right)^2\right]\right)\\
&\leqslant \exp\left(\frac{c}{2}\,\mathbb{E}\left[\left|\sum_{k=t+1}^{T}\frac{\psi_k(S,z)-\mu_k}{k}\right|\right]\right)\\
&\leqslant \exp\left(\frac{c}{2}\sum_{k=t+1}^{T}\frac{\mathbb{E}\left[|\psi_k(S,z)-\mu_k|\right]}{k}\right)\\
&\leqslant \exp\left(c\sum_{k=t+1}^{T}\frac{\mu_k}{k}\right)\,.
\end{aligned}$$

Getting back to (21) we conclude that

$$\mathop{\mathbb{E}}_{S,z}\left[\exp\left(c\sum_{k=t+1}^{T}\frac{\psi_k(S,z)}{k}\right)\right] \leqslant \exp\left(c\sum_{k=t+1}^{T}\frac{2\mu_k}{k}\right)\,. \tag{22}$$

Next, we give an upper-bound on μ_k, that is $\mu_k \leqslant \min\{\beta, \mathbb{E}_{S,z}[\xi_k(S,z)]\}$. Finally, we bound $\mathbb{E}_{S,z}[\xi_k(S,z)]$ using the second result of Lemma 6, which holds for any $k \in [T]$, to get that $\mu_k \leqslant \gamma$, with γ defined in the statement of the theorem.

3) Tuning of t_0. Now we turn our attention back to (20). Considering that we took an expectation w.r.t. the data, we use (22) and the fact that $\mu_k \leqslant \gamma$ to get that

$$\begin{aligned}
\mathop{\mathbb{E}}_{S,z}[\Delta_T(S,z)] &\leqslant \sum_{t=t_0+1}^{T}\exp\left(2c\gamma\sum_{k=t+1}^{T}\frac{1}{k}\right)\frac{2cL}{mt}\\
&\leqslant \sum_{t=t_0+1}^{T}\exp\left(2c\gamma\ln\left(\frac{T}{t}\right)\right)\frac{2cL}{mt}\\
&= \frac{2cL}{m}\left(T^{2c\gamma}\right)\sum_{t=t_0+1}^{T}t^{-2c\gamma-1}\\
&\leqslant \frac{1}{2c\gamma}\frac{2cL}{m}\left(\frac{T}{t_0}\right)^{2c\gamma}\,.
\end{aligned}$$

Plug the above into (18) to get

$$\mathop{\mathbb{E}}_{S,z}\mathop{\mathbb{E}}_{A}\left[f(\boldsymbol{w}_{S,T},z)-f(\boldsymbol{w}_{S^{(i)},T},z)\right] \leqslant \frac{L^2}{\gamma m}\left(\frac{T}{t_0}\right)^{2c\gamma} + r\frac{t_0}{m}\,. \tag{23}$$

Let $q = 2c\gamma$. Then, setting

$$t_0 = \left(\frac{2cL^2}{r}\right)^{\frac{1}{1+q}} T^{\frac{q}{1+q}}$$

minimizes (23). Plugging t_0 back we get that (23) equals to

$$\frac{1+\frac{1}{q}}{m}\left(2cL^2\right)^{\frac{1}{1+q}}(rT)^{\frac{q}{1+q}}\,.$$

This completes the proof. □

A.3.1 Optimistic Rates for Learning with Non-convex Loss Functions

Next we will prove an optimistic bound based on Theorem 4, in other words, the bound that demonstrates fast convergence rate subject to the vanishing empirical risk. First we will need the following technical statement.

Lemma 7. *[7, Lemma 7.2] Let* $c_1, c_2, \dots, c_l > 0$ *and* $s > q_1 > q_2 > \dots > q_{l-1} > 0$. *Then the equation*

$$x^s - c_1 x^{q_1} - c_2 x^{q_2} - \cdots - c_{l-1} x^{q_{l-1}} - c_l = 0$$

has a unique positive solution $x^\star$. *In addition,*

$$x^\star \leqslant \max\left\{ (lc_1)^{\frac{1}{s-q_1}}, (lc_2)^{\frac{1}{s-q_2}}, \cdots, (lc_{l-1})^{\frac{1}{s-q_{l-1}}}, (lc_l)^{\frac{1}{s}} \right\}.$$

Next we prove a useful technical lemma similarly as in [25, Lemma 7].

Lemma 8. *Let* $a, c > 0$ *and* $0 < \alpha < 1$. *Then the inequality*

$$x - ax^\alpha - c \leqslant 0$$

implies

$$x \leqslant \max\left\{ 2^{\frac{\alpha}{1-\alpha}} a^{\frac{1}{1-\alpha}}, (2c)^\alpha a \right\} + c\,.$$

Proof. Consider a function $h(x) = x - ax^\alpha - c$. Applying Lemma 7 with $s = 1$, $l = 2$, $c_1 = a$, $c_2 = c$, and $q_1 = \alpha$ we get that $h(x) = 0$ has a unique positive solution $x^\star$ and

$$x^\star \leqslant \max\left\{ (2a)^{\frac{1}{1-\alpha}}, 2c \right\}\ . \tag{24}$$

Moreover, the inequality $h(x) \leqslant 0$ is verified for $x = 0$, and $\lim_{x \to +\infty} h(x) = +\infty$, so we have that $h(x) \leqslant 0$ implies $x \leqslant x^\star$. Now, using this fact and the fact that $h(x^\star) = 0$, we have that

$$x \leqslant x^\star = a\,(x^\star)^\alpha + c\,,$$

and upper-bounding $x^\star$ by (24) we finally have

$$x \leqslant a \max\left\{ (2a)^{\frac{\alpha}{1-\alpha}}, (2c)^\alpha \right\} + c\,,$$

which completes the proof. □

Proof of Corollary 2. Consider Theorem 4 and observe that it verifies condition of Lemma 8 with $x = \mathbb{E}_{S,A}[R(A_S)]$, $c = \mathbb{E}_{S,A}\left[\widehat{R}_S(A_S)\right]$, $\alpha = \frac{c\gamma}{1+c\gamma}$, and

$$a = \frac{1 + \frac{1}{c\gamma}}{m} \left(2cL^2\right)^{\frac{1}{1+c\gamma}} T^{\frac{c\gamma}{1+c\gamma}}\ .$$

Note that $\alpha/(1-\alpha) = c\gamma$ and $1/(1-\alpha) = 1 + c\gamma$. Then, we obtain that

$$\begin{aligned}
&\mathop{\mathbb{E}}_{S,A}\left[R(A_S) - \widehat{R}_S(A_S)\right] \\
&\leqslant \max\left\{ 2^{c\gamma}\left(\frac{1+\frac{1}{c\gamma}}{m}\right)^{1+c\gamma} \left(2cL^2\right) T^{c\gamma}, \left(2 \mathop{\mathbb{E}}_{S,A}\left[\widehat{R}_S(A_S)\right]\right)^{\frac{c\gamma}{1+c\gamma}} \left(\frac{1+\frac{1}{c\gamma}}{m}\left(2cL^2\right)^{\frac{1}{1+c\gamma}} T^{\frac{c\gamma}{1+c\gamma}}\right) \right\} \\
&= \max\left\{ \left(2 + \frac{2}{c\gamma}\right)^{1+c\gamma} \left(cL^2\right) \left(\frac{T^{c\gamma}}{m^{1+c\gamma}}\right), \frac{1+\frac{1}{c\gamma}}{m} \left(2cL^2\right)^{\frac{1}{1+c\gamma}} \left(2 \mathop{\mathbb{E}}_{S,A}\left[\widehat{R}_S(A_S)\right] \cdot T\right)^{\frac{c\gamma}{1+c\gamma}} \right\}.
\end{aligned}$$

This completes the proof. □

Proof of Proposition 1. Consider minimizing the bound given by Corollary 1 (in the submission file) over a discrete set of source hypotheses $\{\boldsymbol{w}_k^{\mathrm{src}}\}_{k=1}^K$,

$$\begin{aligned} &\min_{k\in[K]} \epsilon(\mathcal{D}, \boldsymbol{w}_k^{\mathrm{src}}) \\ &\leqslant \min_{k\in[K]} \mathcal{O}\left(\frac{1+\frac{1}{c\gamma_k}}{m}\left(R(\boldsymbol{w}_k^{\mathrm{src}})\cdot T\right)^{\frac{c\gamma_k}{1+c\gamma_k}}\right), \end{aligned} \tag{25}$$

and let

$$\begin{aligned} \gamma_k &= \mathcal{O}\left(\mathop{\mathbb{E}}_{z\sim\mathcal{D}}\left[\|\nabla^2 f(\boldsymbol{w}_k^{\mathrm{src}}, z)\|_2\right] + \sqrt{R(\boldsymbol{w}_k^{\mathrm{src}})}\right), \\ \widehat{\gamma}_k &= \frac{1}{m}\sum_{i=1}^m \|\nabla^2 f(\boldsymbol{w}_k^{\mathrm{src}}, z_i)\|_2 + \sqrt{\widehat{R}_S(\boldsymbol{w}_k^{\mathrm{src}})}\,. \end{aligned}$$

By Hoeffding inequality, with high probability, we have that $|\gamma_k - \widehat{\gamma}_k| \leqslant \mathcal{O}\left(\frac{1}{\sqrt[4]{m}}\right)$. Now we further upper bound (25) by upper bounding $R(\boldsymbol{w}_k^{\mathrm{src}})$ and apply union bound to get

$$\begin{aligned} &\min_{k\in[K]} \epsilon(\mathcal{D}, \boldsymbol{w}_k^{\mathrm{src}}) \\ &\leqslant \min_{k\in[K]} \mathcal{O}\left(\left(1+\frac{1}{c\widehat{\gamma}_k^-}\right)\widehat{R}_S(\boldsymbol{w}_k^{\mathrm{src}})^{\frac{c\widehat{\gamma}_k^+}{1+c\widehat{\gamma}_k^+}}\cdot\frac{\sqrt{\log(K)}}{m^{\frac{1}{1+c\widehat{\gamma}_k^+}}}\right), \end{aligned}$$

where $\widehat{\gamma}_k^{\pm} = \widehat{\gamma}_k \pm \frac{1}{\sqrt[4]{m}}$. This completes the proof. □