Stability Analysis and Generalization Bounds of Adversarial Training

Jiancong Xiao[1,*], **Yanbo Fan**[2,†], **Ruoyu Sun**[1,3,†], **Jue Wang**[2], **Zhi-Quan Luo**[1,3]
[1]The Chinese University of Hong Kong, Shenzhen;
[2]Tencent AI Lab; [3]Shenzhen Research Institute of Big Data
jiancongxiao@link.cuhk.edu.cn, fanyanbo0124@gmail.com,
sunruoyu@cuhk.edu.cn, arphid@gmail.com, luozq@cuhk.edu.cn

Abstract

In adversarial machine learning, deep neural networks can fit the adversarial examples on the training dataset but have poor generalization ability on the test set. This phenomenon is called robust overfitting, and it can be observed when adversarially training neural nets on common datasets, including SVHN, CIFAR-10, CIFAR-100, and ImageNet. In this paper, we study the robust overfitting issue of adversarial training by using tools from uniform stability. One major challenge is that the outer function (as a maximization of the inner function) is nonsmooth, so the standard technique (*e.g.,* (Hardt et al., 2016)) cannot be applied. Our approach is to consider η-approximate smoothness: we show that the outer function satisfies this modified smoothness assumption with η being a constant related to the adversarial perturbation ϵ. Based on this, we derive stability-based generalization bounds for stochastic gradient descent (SGD) on the general class of η-approximate smooth functions, which covers the adversarial loss. Our results suggest that robust test accuracy decreases in ϵ when T is large, with a speed between $\Omega(\epsilon\sqrt{T})$ and $\mathcal{O}(\epsilon T)$. This phenomenon is also observed in practice. Additionally, we show that a few popular techniques for adversarial training (*e.g.,* early stopping, cyclic learning rate, and stochastic weight averaging) are stability-promoting in theory.

1 Introduction

Deep neural networks (DNNs) (Krizhevsky et al., 2012; Hochreiter and Schmidhuber, 1997) have become successful in many machine learning tasks and rarely suffered overfitting issues (Zhang et al., 2021). A neural network model can be trained to achieve zero training error and generalize well to the unseen data. While in the setting of adversarial training, robust overfitting is a dominant issue (Rice et al., 2020). Specifically, robust overfitting characterizes a training procedure shown in Fig. 1. After a particular epoch, the robust test accuracy (black line) starts to decrease, but the robust training accuracy (blue line) is still increasing. This phenomenon can be observed in the experiments on common datasets, *e.g.,* SVHN, CIFAR-10, CIFAR-100, and ImageNet. Recent works (Gowal et al., 2020; Rebuffi et al., 2021) mitigated the overfitting issue using regularization techniques such as stochastic weight averaging (SWA)

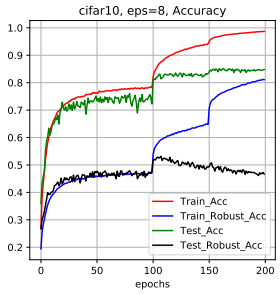


Figure 1: Experiments of adversarial training on CIFAR-10.

[*]This work is done when Jiancong Xiao is a research intern at Tencent AI Lab, China.
[†]Corresponding Authors.

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

and early stopping, but it still has a large generalization gap. Therefore, it is essential to study this issue from a theoretical perspective. In this paper, we study the robust overfitting issue of adversarial training by using tools from uniform stability.

Uniform stability analysis (Bousquet and Elisseeff, 2002) in learning problems has been introduced to measure generalization gap instead of uniform convergence analysis such as classical VC-dimension (Vapnik and Chervonenkis, 2015) and Rademacher complexity (Bartlett and Mendelson, 2002). Generalization gap can be bounded in terms of uniform argument stability (UAS). Formally, UAS is the gap between the output parameters θ of running an algorithm A on two datasets S and S' differ in at most one sample, denoted as $\delta(S, S') = \|\theta(S) - \theta(S')\|$. In a standard training problem with n training samples, assuming that the loss function is convex, L-Lipschitz and β-gradient Lipschitz, running stochastic gradient descent (SGD) with step size $\alpha \leq 1/\beta$ for T steps, the UAS is bounded by $\mathcal{O}(L\alpha T/n)$ (Hardt et al., 2016). Since the generalization gap is controlled by the number of samples n, this bound (at least partially) explains the good generalization of a standard training problem. Beyond (Hardt et al., 2016), Bassily et al. (2020) considered the case that the loss function is non-smooth. Without the β-gradient Lipschitz assumption, they showed that the UAS is bounded by $\mathcal{O}(L\alpha\sqrt{T} + L\alpha T/n)$ and provided a lower bound to show that the additional term $\mathcal{O}(L\alpha\sqrt{T})$ is unavoidable. In adversarial settings, two works have discussed the stability of adversarial training to our knowledge.

Firstly, Farnia and Ozdaglar (2021) considered the stability of minimax problems. Their work includes a discussion on an algorithm called GDmax (gradient descent on the maximization of the inner problem), which can be viewed as a general form of adversarial training. It provides the stability of GDmax when the inner problem is further assumed to be strongly concave. Under the strongly concave assumption, the outer function is smooth. Then, the generalization bound $\mathcal{O}(L\alpha T/n)$ can be applied in this case. However, the inner problem is not strongly concave in practice. The bound $\mathcal{O}(L\alpha T/n)$ does not match the poor generalization of adversarial training observed in practice.

Another stability analysis of adversarial training is the work of (Xing et al., 2021a). Before they propose their algorithm, they use the stability bound $\mathcal{O}(L\alpha\sqrt{T} + L\alpha T/n)$ (Bassily et al., 2020) to characterize the issue of adversarial generalization. However, the bound is ϵ-independent. Let us Consider two cases, $\epsilon \to 0$ and ϵ is large (*e.g.,* 16/255). In the first case, adversarial training is very close to standard training and has good generalization ability. In the second case, adversarial generalization is very poor. The bound is the same in these two different cases. Therefore, it cannot provide an interpretation of adversarial generalization.

In summary, existing stability-based generalization bounds (Hardt et al., 2016; Bassily et al., 2020) have limited interpretation on adversarial generalization. In this paper, we provide a new stability analysis of adversarial training using the notion of η-approximate smoothness. We first show that, under the same assumptions as (Hardt et al., 2016; Xing et al., 2021a), even though the outer function (adversarial loss) is non-smooth, it is η-approximately smooth (see Definition 4.1), where η is a constant linearly depending on the gradient Lipschitz of the inner function and the perturbation intensity ϵ. Then, we derive stability-based generalization bounds (Thm. 5.1 to 5.4) for stochastic gradient descent (SGD) on this general class of η-approximate smooth functions, which covers the adversarial loss. Our main result can be summarized in the following equation. Running SGD on adversarial loss for T steps with step size $\alpha \leq 1/\beta$, the excess risk, which is the sum of generalization error and optimization error, satisfies

$$\text{Excess Risk} \leq \mathcal{E}_{gen} + \mathcal{E}_{opt} \leq \underbrace{\overbrace{L\eta T\alpha}^{\text{additional}} + \overbrace{\frac{2L^2T\alpha}{n} + \frac{D^2}{T\alpha} + L^2\alpha}^{\text{for standard training}}}_{\text{for adversarial training}}. \tag{1.1}$$

The excess risk of adversarial training has an additional term $L\eta T\alpha$. We also provide a lower bound of UAS of running SGD on adversarial loss. Our results suggest that robust test accuracy decreases in ϵ when T is large, with a speed between $\Omega(\epsilon\sqrt{T})$ and $\mathcal{O}(\epsilon T)$. This phenomenon is also observed in practice. It provides an understanding of robust overfitting from the perspective of uniform stability. Additionally, we show that a few popular techniques for adversarial training (*e.g.,* early stopping, cyclic learning rate, and stochastic weight averaging) are stability-promoting in theory and also empirically improve adversarial training. Experiments on SVHN, CIFAR-10, CIFAR-100, and ImageNet confirm our results. Our contributions are listed as follows:

- Main results: we derive stability-based generalization bounds for adversarial training using the notion of η-approximate smoothness. Based on this, we provide an analysis to understand robust overfitting.
- We provide the stability analysis of a few popular techniques for adversarial training and show that they are indeed stability-promoting.
- We provide experiments on SVHN, CIFAR-10, CIFAR-100, and ImageNet. The results verify the generalization bounds.
- Technical contribution: we develop a set of properties of η-approximately smooth function, which might be useful in other tasks.

The paper is organized as follows. After discussing the related work in Sec. 2, the rest of the paper contains two parts. The first part is the technical part to derive stability bounds. The second part is to analyze robust overfitting. Specifically, in the first part, Sec. 3 introduces the preliminary knowledge about UAS. Sec. 4 provides the Lemma and properties of approximately smooth functions and Sec. 5 gives the stability bounds. In the second part, Sec. 6 analyzes the robust overfitting in the theoretical settings and Sec. 7 presents the experiments.

2 Related Work

Adversarial Attacks and Defense. Starting from the work of (Szegedy et al., 2013), it has been commonly realized that deep neural networks are highly susceptible to imperceptible corruptions to the input data (Goodfellow et al., 2014; Carlini and Wagner, 2017; Madry et al., 2017). A series of work aimed at training neural networks robust to such small perturbations (Wu et al., 2020; Gowal et al., 2020; Zhang et al., 2020) and another line of work aimed at designing more powerful adversarial attack algorithms (Athalye et al., 2018; Tramer et al., 2020; Fan et al., 2020; Xiao et al., 2022c; Qin et al., 2022). A series of work considered adversarial robustness in black-box settings (Chen et al., 2017; Qin et al., 2021). Semi-supervised learning has been used to improve adversarial robustness (Carmon et al., 2019; Li et al., 2022). Fast adversarial training (Wong et al., 2020; Huang et al., 2022) was introduced to save training time.

Adversarial Generalization. A series of work tried to explain adversarial generalization in the uniform convergence framework, including VC-dimension (Attias et al., 2021; Montasser et al., 2019) and Rademacher complexity (Khim and Loh, 2018; Yin et al., 2019; Awasthi et al., 2020; Xiao et al., 2022a). Uniform algorithmic stability is another framework to study adversarial generalization (Farnia and Ozdaglar, 2021; Xing et al., 2021a; Xiao et al., 2022b). The work of (Schmidt et al., 2018; Raghunathan et al., 2019; Zhai et al., 2019) have shown that in some scenarios achieving adversarial generalization requires more data. The work of (Xing et al., 2021b,c; Javanmard et al., 2020) studied the generalization in the setting of adversarial linear regression. (Sinha et al., 2017) studied the generalization of distributional robustness. The work of (Taheri et al., 2020; Javanmard et al., 2020; Dan et al., 2020) analyzed adversarial generalization in Gaussian mixture models.

Uniform Stability. Stability is a classical approach to provide generalization bounds. It can be traced back to the work of (Rogers and Wagner, 1978). After a few decades, it was well developed in analyzing the generalization bounds in statistical learning problems (Bousquet and Elisseeff, 2002). These bounds have been significantly improved in a recent sequence of works (Feldman and Vondrak, 2018, 2019). The work of (Chen et al., 2018) derived minimax lower bounds for excess risk and discussed the optimal trade-off between stability and convergence. Ozdaglar et al. (2022) considered the generalization metric of minimax optimizer.

3 Preliminaries of Stability

Consider the following setting of statistical learning. There is an unknown distribution $\mathcal{D}$ over examples from some space $\mathcal{Z}$. We receive a sample dataset $S = \{z_1, \ldots, z_n\}$ of n examples drawn i.i.d. from $\mathcal{D}$. The *population risk* and *empirical risk* are defined as:

$$R_{\mathcal{D}}(\theta) \stackrel{\text{def}}{=} \mathbb{E}_{z\sim\mathcal{D}}\, h(\theta, z) \quad \text{and} \quad R_S(\theta) \stackrel{\text{def}}{=} \frac{1}{n}\sum_{i=1}^{n} h(\theta, z_i),$$

respectively, where $h(\cdot,\cdot)$ is the loss function.

Risk Decomposition. Let θ^* and $\bar{\theta}$ be the optimal solution of $R_{\mathcal{D}}(\theta)$ and $R_S(\theta)$ respectively. Then for the algorithm output $\hat{\theta} = A(S)$, the excess risk can be decomposed as

$$R_{\mathcal{D}}(\hat{\theta}) - R_{\mathcal{D}}(\theta^*) = \underbrace{R_{\mathcal{D}}(\hat{\theta}) - R_S(\hat{\theta})}_{\mathcal{E}_{gen}} + \underbrace{R_S(\hat{\theta}) - R_S(\bar{\theta})}_{\mathcal{E}_{opt}} + \underbrace{R_S(\bar{\theta}) - R_S(\theta^*)}_{\leq 0} + \underbrace{R_S(\theta^*) - R_{\mathcal{D}}(\theta^*)}_{\mathbb{E}=0}.$$

To control the excess risk, we need to control the generalization gap $\mathcal{E}_{gen}$ and the optimization gap $\mathcal{E}_{opt}$. In the rest of the paper, we use $\mathcal{E}_{gen}$ and $\mathcal{E}_{opt}$ to denote the *expectation* of the generalization and optimization gap. To bound the generalization gap of a model $\hat{\theta} = A(S)$ trained by a randomized algorithm A, we employ the following notion of *uniform stability*.

Definition 3.1. *A randomized algorithm A is ε-*uniformly stable *if for all data sets $S, S' \in \mathcal{Z}^n$ such that S and S' differ in at most one example, we have*

$$\sup_z \mathbb{E}_A \left[h(A(S); z) - h(A(S'); z)\right] \leq \varepsilon\,. \tag{3.1}$$

Here, the expectation is taken only over the internal randomness of A. We recall the important theorem that uniform stability implies *generalization in expectation* (Hardt et al., 2016).

Theorem 3.1 (Generalization in expectation)**.** *Let A be ε-uniformly stable. Then, the expected generalization gap satisfies*

$$|\mathcal{E}_{gen}| = |\mathbb{E}_{S,A}[R_{\mathcal{D}}[A(S)] - R_S[A(S)]]| \leq \varepsilon\,.$$

Therefore, we turn to the properties of iterative algorithms that control their uniform stability.

4 Stability of Adversarial Training

Adversarial Surrogate Loss. In adversarial training, we consider the following surrogate loss

$$h(\theta; z) = \max_{\|z - z'\|_p \leq \epsilon} g(\theta; z'), \tag{4.1}$$

where $g(\theta; z)$ is the loss function of the standard counterpart, $\|\cdot\|_p$ is the ℓ_p-norm, $p \geq 1$. Usually, g can also be written in the form of $\ell(f_\theta(\boldsymbol{x}); y)$, where f_θ is the neural network to be trained and $(\boldsymbol{x}, y)$ is the input-label pair. We assume the loss function g satisfies the following smoothness assumption.

Assumption 4.1. *The function g satisfies the following Lipschitzian smoothness conditions:*

$$\begin{aligned}
&\|g(\theta_1, z) - g(\theta_2, z)\| \leq L\|\theta_1 - \theta_2\|,\\
&\|\nabla_\theta g(\theta_1, z) - \nabla_\theta g(\theta_2, z)\| \leq L_\theta\|\theta_1 - \theta_2\|,\\
&\|\nabla_\theta g(\theta, z_1) - \nabla_\theta g(\theta, z_2)\| \leq L_z\|z_1 - z_2\|_p,
\end{aligned}$$

where $\|\cdot\|$ is Euclidean norm.

Assumption 4.1 assumes that the loss function is smooth, which are also used in the stability literature (Farnia and Ozdaglar, 2021; Xing et al., 2021a), as well as the convergence analysis literature (Wang et al., 2019; Liu et al., 2020). While ReLU activation function is non-smooth, recent works (Allen-Zhu et al., 2019; Du et al., 2019) showed that the loss function of over-parameterized DNNs is semi-smooth. It helps justify Assumption 4.1. Under Assumption 4.1, the loss function of adversarial training satisfies the following Lemma (Liu et al., 2020).

Lemma 4.1. *Let h be the adversarial loss defined in Eq. (4.1) and g satisfies Assumption 4.1. $\forall\theta_1, \theta_2$ and $\forall z \in \mathcal{Z}$, the following properties hold.*

1. *(Lipschitz function.) $\|h(\theta_1, z) - h(\theta_2, z)\| \leq L\|\theta_1 - \theta_2\|$.*
2. *For all subgradient $d(\theta, z) \in \partial_\theta h(\theta, z)$, we have $\|d(\theta_1, z) - d(\theta_2, z)\| \leq L_\theta\|\theta_1 - \theta_2\| + 2L_z\epsilon$.*

If we further assume that $g(\theta, z)$ is μ-strongly concave in z, the adversarial surrogate loss $h(\theta, z)$ is also smooth (Sinha et al., 2017). Therefore, the uniform stability of adversarial training follows (Hardt et al., 2016; Farnia and Ozdaglar, 2021), see Appendix B.3. However, $g(\theta, z)$ is non-strongly concave in practice. The above results provide limited explanations of the poor generalization of adversarial training. We discuss the generalization properties of adversarial training under Lemma 4.1.2.

4.1 Basic Properties of Approximate Smoothness

To simplify the notation, we use $h(\theta)$ as a shorthand notation of $h(\theta, z)$. To simplify the argument, we consider differentiable function h. The results can be extended to non-differentiable cases. Lemma 4.1 motivates us to analyze a function with the following modified smoothness assumption, which we call approximate smoothness assumption.

Definition 4.1. *Let $\beta > 0$ and $\eta > 0$. We say a differentiable function $h(\theta)$ is η-approximately β-gradient Lipschitz, if $\forall \theta_1$ and θ_2, we have*

$$\|\nabla h(\theta_1) - \nabla h(\theta_2)\| \leq \beta \|\theta_1 - \theta_2\| + \eta.$$

In definition 4.1, η controls the smoothness of the loss function $h(\cdot)$. When $\eta = 0$, the function h is gradient Lipschitz. When $\eta \to +\infty$, h is a general non-smooth function. As our discussion before, the adversarial surrogate loss is $2L_z\epsilon$-approximately smooth. As far as we know, this assumption is rarely discussed in the optimization literature since it cannot improve the convergence rate from a general non-smooth assumption. But it affects uniform stability, as we will discuss later. We need to develop the basic properties of approximate smoothness first.

Lemma 4.2. *Assume that the function h is η-approximately β-gradient Lipschitz. $\forall \theta_1, \theta_2$ and $\forall z \in \mathcal{Z}$, the following properties hold.*

1. *(η-approximate descent lemma.)*
$$h(\theta_1) - h(\theta_2) \leq \nabla h(\theta_2)^T(\theta_1 - \theta_2) + \frac{\beta}{2}\|\theta_1 - \theta_2\|^2 + \eta\|\theta_1 - \theta_2\|.$$
2. *(η-approximately co-coercive.) Assume in addition that $h(\theta, z)$ is convex in θ for all $z \in \mathcal{Z}$. Let $[\cdot]_+ = \max(0, \cdot)$. We have*
$$\langle \nabla h(\theta_1) - \nabla h(\theta_2), \theta_1 - \theta_2 \rangle \geq \frac{1}{\beta}\Big[[\|\nabla h(\theta_1) - \nabla h(\theta_2)\| - \eta]_+\Big]^2.$$

We defer the proof to Appendix A. Note that the loss function is L-Lipschitz for every example z, we have $\mathbb{E}|h(\theta_1, z) - h(\theta_2, z)| \leq L\mathbb{E}\|\theta_1 - \theta_2\|$, for all $z \in \mathcal{Z}$. To obtain the stability generalization bounds, we need to analysis the difference $\|\theta_1^T - \theta_2^T\|$, where θ_1^T and θ_2^T are the outputs of running SGD on adversarial surrogate loss for T iterations on two datasets with only one different sample. Next we provide the recursive bounds under the approximate smoothness assumption.

Algorithms. We consider the stochastic gradient descent on the adversarial surrogate loss. i.e.,

$$\theta^{t+1} = \theta^t - \alpha_t \nabla_\theta h(\theta^t, z_{i_t}), \tag{4.2}$$

where α_t is the step size in iteration t, z_{i_t} is the sample chosen in iteration t. We consider two popular schemes for choosing the examples indices i_t. *Sampling with replacement*: One is to pick $i_t \sim \text{Uniform}\{1, \cdots, n\}$ at each step. *Fixed permutation*: The other is to choose a random permutation over $\{1, \cdots, n\}$ and cycle through the examples repeatedly in the order determined by the permutation. Our results hold for both variants.

Properties of Update Rules. We define $G_{\alpha,z}(\theta) = \theta - \alpha\nabla h(\theta, z)$ be the update rule of SGD. The following lemma holds.

Lemma 4.3. *Assume that the function h is η-approximately β-gradient Lipschitz. $\forall \theta_1, \theta_2$ and $\forall z \in \mathcal{Z}$,we have*

1. *($\alpha\eta$-approximately $(1 + \alpha\beta)$-expansive.) $\|G_{\alpha,z}(\theta_1) - G_{\alpha,z}(\theta_2)\| \leq (1 + \alpha\beta)\|\theta_1 - \theta_2\| + \alpha\eta$.*
2. *($\alpha\eta$-approximately non-expansive.) Assume in addition that $h(\theta, z)$ is convex in θ for all $z \in \mathcal{Z}$, for $\alpha \leq 1/\beta$, we have $\|G_{\alpha,z}(\theta_1) - G_{\alpha,z}(\theta_2)\| \leq \|\theta_1 - \theta_2\| + \alpha\eta$.*
3. *($\alpha\eta$-approximately $(1 - \alpha\gamma)$-contractive.) Assume in addition that $h(\theta, z)$ is γ-strongly convex in θ for all $z \in \mathcal{Z}$, for $\alpha \leq 1/\beta$, we have $\|G_{\alpha,z}(\theta_1) - G_{\alpha,z}(\theta_2)\| \leq (1 - \alpha\gamma)\|\theta_1 - \theta_2\| + \alpha\eta$.*

The proof of Lemma 4.3 is based on Lemma 4.2 and is deferred to Appendix A. Lemma 4.3 provides the recursive distance $\|\theta_1 - \theta_2\|$ from iteration t to $t + 1$. Based on this, we can recursively derive the distance $\|\theta_1^T - \theta_2^T\|$. Then, we can obtain the stability generalization bounds.

5 Stability Generalization Bounds

In the previous section, we have discussed the properties of approximate smoothness and developed tools we need to use. In this section, we discuss the stability bounds.

5.1 Convex Optimization

We first consider the case that $h(\theta, z)$ is convex in θ for all $z \in \mathcal{Z}$.

Theorem 5.1. *Assume that $h(\theta, z)$ is convex, L-Lipschitz, and η-approximately β-gradient Lipschitz in θ for all given $z \in \mathcal{Z}$. Suppose that we run SGD with step sizes $\alpha_t \leq 1/\beta$ for T steps. Then,*

$$\mathcal{E}_{gen} = \mathbb{E}[R_{\mathcal{D}}(\theta^T) - R_S(\theta^T)] \leq L\left(\eta + \frac{2L}{n}\right)\sum_{t=1}^{T}\alpha_t. \tag{5.1}$$

The proof is mainly based on Lemma 4.3 that the update rule is approximately non-expansive in this case. We defer it to appendix A. We have shown that the adversarial surrogate loss is $2L_z\epsilon$-approximately L_θ-gradient Lipschitz. Let $\eta = 2L_z\epsilon$ in Eq. (5.1), we directly obtain the stability bounds for adversarial training. In practice, the solution of the inner problem is sub-optimal. Let $\Delta\epsilon$ be the maximum error between the optimal and sub-optimal attacks in each iteration. We have the following Corollary.

Corollary 5.1 (Uniform stability for sub-optimal attacks adversarial training.)**.** *Under Assumption 4.1, assume in addition that $g(\theta, z)$ is convex in θ for all given $z \in \mathcal{Z}$. Suppose that we run adversarial training with step sizes $\alpha_t \leq 1/L_\theta$ for T steps. Then, adversarial training satisfies uniform stability with*

$$\mathcal{E}_{gen} \leq \left(2LL_z(\epsilon + \Delta\epsilon) + \frac{2L^2}{n}\right)\sum_{t=1}^{T}\alpha_t \leq \mathcal{O}\left(L(L_z\epsilon + \frac{L}{n})\sum_{t=1}^{T}\alpha_t\right). \tag{5.2}$$

Remark: Corollary 5.1 shows that adversarial training with weak attacks (large $\Delta\epsilon$) have worse adversarial generalization than that with strong attacks. This is also observed in practice. However, $\Delta\epsilon$ is at most 2ϵ, the upper bound of AT with different attacks have the same order. It might be due to the weakness of Assumption 4.1 or uniform stabilty framework.

Interpreting Robust Generalization. If we let $\epsilon = 0$ in Eq. (5.2) , it reduced to the generalization bound in (Hardt et al., 2016) for standard training. Therefore, the additional generalization error of adversarial training comes from the additional term $L_z\epsilon$. The global gradient Lipschitz L_z with respect to z plays an important role in generalization. Even though the perturbation ϵ is small, it is amplified by the Lipschitz L_z and finally hurts the robust generalization.

5.2 Further Discussion on the Generalization Bounds

We first provide a lower bound. Then we compare our bounds with the existing bounds in Table 1.

Theorem 5.2 (Lower Bound)**.** *There exists functions $h(\theta, z)$, s.t. h is convex, L-Lipschitz, and η-approximately β-gradient Lipschitz in θ for all given $z \in \mathcal{Z}$. Exists S and S' differ in one sample. Suppose that we run SGD with step fixed step sizes $\alpha \leq 1/\beta$ for T steps. Then,*

$$\mathbb{E}[\delta(S, S')] \geq \Omega\left(\eta\alpha\sqrt{T} + \frac{L\alpha T}{n}\right). \tag{5.3}$$

Table 1: Comparison of the upper and lower bounds of $\mathbb{E}[\delta(S, S')]$. Comparing with the previous results, we replace L by η and provide the matching lower bound in η.

	Assumption	Upper bounds	Lower bounds
Farnia and Ozdaglar (2021)	convex-strongly concave	$\mathcal{O}(LT\alpha/n)$	$\Omega(LT\alpha/n)$
Xing et al. (2021a)	convex-nonconcave	$\mathcal{O}(L\sqrt{T}\alpha + LT\alpha/n)$	$\Omega(\sqrt{T}\alpha + LT\alpha/n)$
Ours	convex-nonconcave	$\mathcal{O}(\eta T\alpha + LT\alpha/n)$	$\Omega(\eta\sqrt{T}\alpha + LT\alpha/n)$

Comparison with the Existing UAS Bounds. Compared with the work of (Farnia and Ozdaglar, 2021), they assume that the inner problem is strongly concave. Thus the bounds are not comparable.

Strongly concave is a strong assumption in practice. Therefore, the work of (Xing et al., 2021a) and our analysis focus on the nonconcave cases. Comparing with the bound $\mathcal{O}(L\sqrt{T}\alpha + LT\alpha/n)$, our bound captures a critical aspect of adversarial generalization bound: ϵ-dependent. As observed in practice, the adversarial generalization gap reduces to the standard generalization gap as $\epsilon \to 0$. Our bound consists with this observation. On the contrary, the bound $\mathcal{O}(L\sqrt{T}\alpha + LT\alpha/n)$ is very large when $\epsilon \to 0$. Additionally, we provide a matching lower bound w.r.t η. The comparison is provided in Table 1.

Comparison with the Work of (Xing et al., 2021a). The work of (Xing et al., 2021a) argued that the max function is not smooth even though the standard counterpart is smooth. Therefore, they followed the bound in non-smooth cases (Bassily et al., 2020). Then, they aimed to solve the non-smooth issue. They design a noise-injected algorithm and show its effectiveness in tackling the non-smooth issue. Our work focus on providing better bounds to interpret robust overfitting.

5.3 Non-convex Optimization and Strongly Convex Optimization

Next, we consider the case that the loss function h is general non-convex and strongly convex. By Lemma 4.3, we have

Theorem 5.3. *Assume that $h(\theta, z)$ is L-Lipschitz, and η-approximately β-gradient Lipschitz in θ for all given $z \in \mathcal{Z}$. Assume in addition that $0 \leq h(\theta, z) \leq B$ for all θ and z. Suppose that we run SGD with diminishing step sizes $\alpha_t \leq 1/(\beta t)$ for T steps. Then*

$$\mathcal{E}_{gen} \leq \frac{BL_\theta + (2L^2 + L\eta n)T}{\beta(n-1)}. \tag{5.4}$$

Theorem 5.4. *Assume that $h(\theta, z)$ is γ-strongly convex, L-Lipschitz, and η-approximately β-gradient Lipschitz in θ for all given $z \in \mathcal{Z}$. Suppose that we run SGD with step sizes $\alpha_t \leq 1/\beta$ for T steps. Then*

$$\mathcal{E}_{gen} = \mathbb{E}[R_{\mathcal{D}}(\theta^T) - R_S(\theta^T)] \leq \frac{L\eta}{\gamma} + \frac{2L^2}{\gamma n}. \tag{5.5}$$

Remark: The bound in non-convex cases provides a similar interpretation of robust generalization to the analysis in the convex case. In uniform stability analysis, whether the loss function is convex or non-convex does not give a major difference. Therefore, we provide the analysis of the non-convex case in Appendix B. We also provide our convergence analysis of running SGD on a η-approximate smoothness, non-convex function in Theorem B.1. Strongly convex is a strong assumption. We leave the analysis of the bound in strongly convex cases in Appendix B.2.

6 Excess Risk Minimization

Based on the risk decomposition, we have Excess Risk $\leq \mathcal{E}_{gen} + \mathcal{E}_{opt}$, we need to minimize $\mathcal{E}_{gen} + \mathcal{E}_{opt}$ to achieve better performance. Per our previous discussion, whether the loss function is convex or non-convex does not give a major difference in stability analysis. We study the convex case in this section. We leave the discussion on the non-convex and strongly convex cases in Appendix B.1 and B.2, respectively. We first introduce the optimization error.

Optimization Analysis. The convergence analysis of SGD on a L-Lipschitz, convex function is discussed in (Nemirovski et al., 2009). The convergence rate cannot be improved if we further assume that the function h is gradient Lipschitz. Therefore, a weaker condition, approximately gradient Lipschitz, cannot improve the convergence rate. We use the following convergence error bound (adopted from (Nemirovski et al., 2009)) for the optimization error of both adversarial training and standard training.

Theorem 6.1. *Assume that $h(\theta, z)$ is L-Lipschitz and convex in θ for all given $z \in \mathcal{Z}$. Let $D = \|\theta^0 - \theta^*\|$, where θ^0 is the initialization of SGD. Suppose that we run SGD with step sizes α_t for T steps. Then, $\exists k \leq T$, s.t.*

$$\mathcal{E}_{opt}(\theta^k) \leq \frac{D^2 + L^2 \sum_{t=1}^T \alpha_t^2}{\sum_{t=1}^T \alpha_t}. \tag{6.1}$$

If we let $\alpha_t = 1/\sqrt{T}$, we have $\mathcal{E}_{opt} \leq \mathcal{O}(1/\sqrt{T})$, which is the convergence rate of SGD on convex function. Since we need to consider the generalization and optimization errors simultaneously, we keep the α_t in Theorem 6.1.

Generalization-Optimization Trade-off. We have now discussed the optimization and generalization errors of adversarial training. We aim to find the optimal trade-off between generalization and optimization in terms of α_t and T. However, finding the optimal α_t and T simultaneously is challenging. We consider the settings we use in practice.

Fixed Step Size. We first consider the simplest case, the step size α is fixed. Then, combining Eq. (5.1) and Eq. (6.1), we have

$$\mathcal{E}_{gen} + \mathcal{E}_{opt} \leq \underbrace{\overbrace{L\eta T\alpha}^{additional} + \overbrace{\frac{2L^2 T\alpha}{n} + \frac{D^2}{T\alpha} + L^2\alpha}^{\text{for standard training}}}_{\text{for adversarial training}}. \tag{6.2}$$

Interpretation of Robust Overfitting. In standard training, overfitting is rarely observed in practice. The optimization error and generalization error are both small. In Eq. (6.2), the second to the fourth terms are for standard training. The second term is controlled by the number of samples n, which is small if we have sufficient training samples. The third term is controlled by T, and the last term is fixed given a small α. This bound partially explains the good performance of standard training. However, we have an additional term $L\eta T\alpha$ for excess risk in adversarial training. Then, after a particular iteration that $L\eta T\alpha$ dominates Eq. (6.2), robust overfitting appears. This is consistent with the training procedure in practice. Therefore, the $\eta = 2L_z\epsilon$ approximate smoothness of adversarial loss provides a possible explanation of robust overfitting. To achieve better performance, we need to stop training the model earlier.

Early Stopping. It is shown that early stopping is an important training technique for adversarial training (Rice et al., 2020). In Eq. (6.2), if we optimize the right-hand-side with respect to T, we have

$$T^* = \frac{\|\theta^0 - \theta^*\|}{\alpha\sqrt{L\eta + 2L^2/n}}, \quad \mathcal{E}_{gen} + \mathcal{E}_{opt} \leq 2\sqrt{L\eta + \frac{2L^2}{n}}D + L^2\alpha.$$

Therefore, it is the best to stop training at T^*. However, T^* is unknown in practice. It is important to select stopping criteria. For example, we can use a validation set to determine when to stop.

Varying Step Size. We discuss one popular varying step size schedule, cyclic learning rate, which is also called super-converge learning rate for adversarial training. It is shown that it can speed up adversarial training with fewer epochs (Wong et al., 2020). The Super-converge learning rate follows the following rules. In the first phase (warm-up), the step size increase from 0 to α' linearly. In the second phase (cold down), the step size decreases back to 0 linearly. It is unclear (to our knowledge) why this schedule can speed up convergence in optimization theory. But the generalization part can partially be explained by UAS. If we set $\alpha' = 2\alpha$, it is easy to check that $\mathcal{E}_{gen} \leq (L\eta + 2L^2/n)T\alpha$ in this case, which is the same as the bound in the fixed learning rate case. Notice that the cyclic learning rate usually requires fewer steps T to converge. Then, the generalization gap is smaller. Cyclic learning rate can be viewed as another form of early stopping from the perspective of UAS.

Stochastic Weight Averaging. SWA is also a useful training technique for adversarial training (Hwang et al., 2021). Instead of using the last checkpoint, SWA suggests using the average of the checkpoints for inference. It is shown that SWA can find a model with better generalization since it leads to wider minima (Izmailov et al., 2018). Below we study SWA from the perspective of UAS.

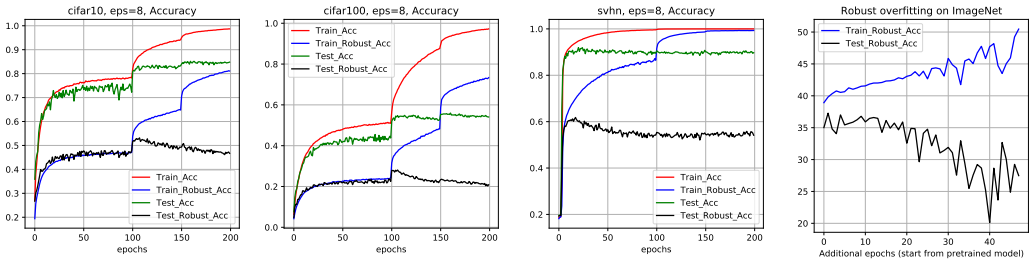
Theorem 6.2. *Assume that $h(\theta, z)$ is convex, L-Lipschitz, and η-approximately β-gradient Lipschitz in θ for all given $z \in \mathcal{Z}$. Suppose that we run SGD with step sizes $\alpha_t \leq 1/\beta$ for T steps. Let $\bar{\theta}$ be the average of the trajectory. Then,*

$$\mathcal{E}_{gen}(\bar{\theta}) \leq \left(\frac{L\eta}{2} + \frac{L^2}{n}\right)\sum_{t=1}^{T}\alpha_t, \quad \mathcal{E}_{opt}(\bar{\theta}) \leq \frac{\|\theta^0 - \theta^*\|^2 + L^2\sum_{t=1}^{T}\alpha_t^2}{\sum_{t=1}^{T}\alpha_t}. \tag{6.3}$$

In words, SWA reduces the generalization error bound to one-half of the one without SWA. But the training error bound remains unchanged.

7 Experiments

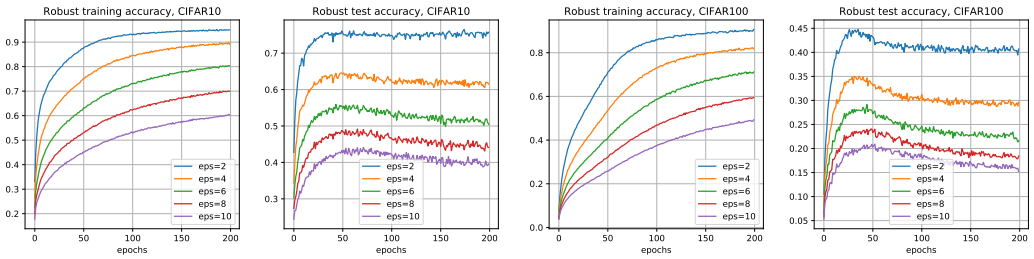
Training Settings. We mainly consider the experiments on CIFAR-10 (Krizhevsky et al., 2009), CIFAR-100, and SVHN (Netzer et al., 2011). We also provide one experiment on ImageNet (Deng et al., 2009). For the first three datasets, we conduct the experiments on training PreActResNet-18, which follows (Rice et al., 2020), For the experiment on ImageNet, we use ResNet-50 (He et al., 2016), following the experiment of (Madry et al., 2017). For the inner problems, we adopt the ℓ_∞ PGD adversarial training in (Madry et al., 2017), the step size in the inner maximization is set to be $\epsilon/4$ on CIFAR-10 and CIFAR100 and is set to be $\epsilon/8$ on SVHN. Weight decay is set to be 5×10^{-4}. Additional experiments are provided in Appendix C. [3]



(a) (b) (c) (d)

Figure 2: Robust overfitting in the experiments on (a) CIFAR-10, (b) CIFAR-10, (c) SVHN and (d) ImageNet.

Robust Overfitting on Common Dataset. In Fig. 2 (a), (b), and (c), we show the experiments on the piece-wise learning rate schedule, which is 0.1 over the first 100 epochs, down to 0.01 over the following 50 epochs, and finally be 0.001 in the last 50 epochs, on CIFAR-10, CIFAR-100, and SVHN. Experiments on different ϵ are shown in Appendix C. *Robust Overfitting on ImageNet.* We provide one experiment on ImageNet in Fig. 2 (d). We start from a pre-trained model from Madry's Lab and keep running 50 more epochs. *Robust overfitting* can be observed in these experiments. After a particular epoch (around the 100th epoch), the robust training accuracy is still increasing, but the robust test accuracy starts to decrease.



(a) (b) (c) (d)

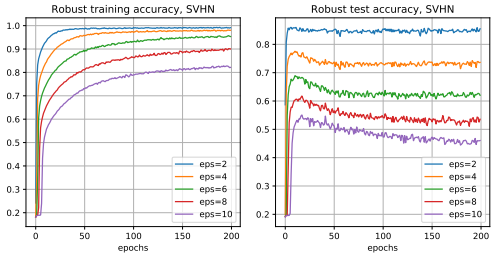
Figure 3: Experiments of adversarial training on CIFAR-10 and CIFAR-100 with a fixed learning rate. (a) Robust training accuracy on CIFAR-10. (b) Robust test accuracy on CIFAR-10. (c) Robust training accuracy on CIFAR-100. (d) Robust test accuracy on CIFAR-100. ϵ are set to be 2, 4, 6, 8, and 10.

[3] `https://github.com/JiancongXiao/Stability-of-Adversarial-Training`

Fixed Step Size. To better understand robust overfitting and match the theoretical settings (in Eq. (6.2)), we consider the fixed learning rate schedule. In Fig. 3, we show the experiments of adversarial training using a fixed learning rate 0.01. The perturbation intensity ϵ is set to be 2, 4, 6, 8, and 10. respectively. Fig. 3 (a) and (b) show the experiments on CIFAR-10. Fig. 3 (c) and (d) show the experiments on CIFAR-100, respectively. Fig. 4 shows the experiments on SVHN.

Generalization Error Dominates Training Error. In the robust overfitting phase, the robust generalization error dominates the robust training error. This phenomenon corresponds to the Eq. (6.2) that the first term dominates the other terms with large T.

Robust Test Accuracy Decreases in ϵ. Comparing different ϵ in Fig. 3, we can see that the robust test accuracy decreases faster when ϵ is larger. This corresponds to the robust overfitting rule in the theoretical setting that the test performance decreases in ϵ. These phenomena are similar in theoretical and practical settings. Therefore, the stability analysis provides a different perspective on understanding robust overfitting.



(a) (b)

Figure 4: Experiments of adversarial training on SVHN with a fixed learning rate. (a) Robust training accuracy. (b) Robust test accuracy.

Robust Test Accuracy Decreases in a Rate between $\Omega(\epsilon\sqrt{T})$ and $\mathcal{O}(\epsilon T)$. If ϵ is small, *e.g.*, $\epsilon = 2$, the decrease rate is close to $\mathcal{O}(\epsilon T)$. If ϵ is large, the decrease rate is more likely to be $\Omega(\epsilon\sqrt{T})$. This is also the gap between the upper bound and lower bound in Sec. 5.

8 Conclusion

Limitations and Future Work. Firstly, in Fig. 3, we can see that the decrease rate of robust overfitting is close to $\mathcal{O}(T)$ when ϵ is small and is close to $\Omega(\sqrt{T})$ when ϵ is large. One possible direction is to figure out the relation. Secondly, one might improve adversarial training by controlling L_z. L_z depends on the loss, the network architecture, and the dataset. One possible direction is to design a smoother loss or smooth activation function (e.g., SiLU) for a lower L_z. Notice that L_z is uniform for all θ. If we view $L_z(\theta)$ locally with respect to θ, we might use an (approximated) second-order penalty term on its magnitude to control it.

In this paper, we show that the adversarial loss satisfies η-approximate smoothness, and we derive stability-based generalization bounds on this general class of η-approximate smooth functions. Our bounds give a different perspective on understanding robust overfitting. The robust test accuracy decreases in η, and experimental results confirm this phenomenon. We think our work will inspire more theoretical and empirical research to improve adversarial training.

Acknowledgement

We would like to thank Jiawei Zhang, Congliang Chen, and Zeyu Qin for the helpful discussions. We thank all the anonymous reviewers for their comments and suggestions. The work of Zhi-Quan Luo is supported by the National Natural Science Foundation of China (No. 61731018) and the Guangdong Provincial Key Laboratory of Big Data Computing.

References

Allen-Zhu, Z., Li, Y., and Song, Z. (2019). A convergence theory for deep learning via over-parameterization. In *International Conference on Machine Learning*, pages 242–252. PMLR.

Athalye, A., Carlini, N., and Wagner, D. (2018). Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. *arXiv preprint arXiv:1802.00420*.

Attias, I., Kontorovich, A., and Mansour, Y. (2021). Improved generalization bounds for adversarially robust learning.

Awasthi, P., Frank, N., and Mohri, M. (2020). Adversarial learning guarantees for linear hypotheses and neural networks. In *International Conference on Machine Learning*, pages 431–441. PMLR.

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

Bassily, R., Feldman, V., Guzmán, C., and Talwar, K. (2020). Stability of stochastic gradient descent on nonsmooth convex losses. *arXiv preprint arXiv:2006.06914*.

Bousquet, O. and Elisseeff, A. (2002). Stability and generalization. *The Journal of Machine Learning Research*, 2:499–526.

Carlini, N. and Wagner, D. (2017). Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pages 39–57. IEEE.

Carmon, Y., Raghunathan, A., Schmidt, L., Duchi, J. C., and Liang, P. S. (2019). Unlabeled data improves adversarial robustness. In *Advances in Neural Information Processing Systems*, pages 11190–11201.

Chen, P.-Y., Zhang, H., Sharma, Y., Yi, J., and Hsieh, C.-J. (2017). Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pages 15–26.

Chen, Y., Jin, C., and Yu, B. (2018). Stability and convergence trade-off of iterative optimization algorithms. *arXiv preprint arXiv:1804.01619*.

Dan, C., Wei, Y., and Ravikumar, P. (2020). Sharp statistical guaratees for adversarially robust gaussian classification. In *International Conference on Machine Learning*, pages 2345–2355. PMLR.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee.

Du, S., Lee, J., Li, H., Wang, L., and Zhai, X. (2019). Gradient descent finds global minima of deep neural networks. In *International Conference on Machine Learning*, pages 1675–1685. PMLR.

Fan, Y., Wu, B., Li, T., Zhang, Y., Li, M., Li, Z., and Yang, Y. (2020). Sparse adversarial attack via perturbation factorization. In *European conference on computer vision*, pages 35–50. Springer.

Farnia, F. and Ozdaglar, A. (2021). Train simultaneously, generalize better: Stability of gradient-based minimax learners. In *International Conference on Machine Learning*, pages 3174–3185. PMLR.

Feldman, V. and Vondrak, J. (2018). Generalization bounds for uniformly stable algorithms. *arXiv preprint arXiv:1812.09859*.

Feldman, V. and Vondrak, J. (2019). High probability generalization bounds for uniformly stable algorithms with nearly optimal rate. In *Conference on Learning Theory*, pages 1270–1279. PMLR.

Goodfellow, I. J., Shlens, J., and Szegedy, C. (2014). Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Gowal, S., Qin, C., Uesato, J., Mann, T., and Kohli, P. (2020). Uncovering the limits of adversarial training against norm-bounded adversarial examples. *arXiv preprint arXiv:2010.03593*.

Hardt, M., Recht, B., and Singer, Y. (2016). Train faster, generalize better: Stability of stochastic gradient descent. In *International Conference on Machine Learning*, pages 1225–1234. PMLR.

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

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Huang, Z., Fan, Y., Liu, C., Zhang, W., Zhang, Y., Salzmann, M., Süsstrunk, S., and Wang, J. (2022). Fast adversarial training with adaptive step size. *arXiv preprint arXiv:2206.02417*.

Hwang, J.-W., Lee, Y., Oh, S., and Bae, Y. (2021). Adversarial training with stochastic weight average. In *2021 IEEE International Conference on Image Processing (ICIP)*, pages 814–818. IEEE.

Izmailov, P., Podoprikhin, D., Garipov, T., Vetrov, D., and Wilson, A. G. (2018). Averaging weights leads to wider optima and better generalization. *arXiv preprint arXiv:1803.05407*.

Javanmard, A., Soltanolkotabi, M., and Hassani, H. (2020). Precise tradeoffs in adversarial training for linear regression. In *Conference on Learning Theory*, pages 2034–2078. PMLR.

Khim, J. and Loh, P.-L. (2018). Adversarial risk bounds via function transformation. *arXiv preprint arXiv:1810.09519*.

Krizhevsky, A., Hinton, G., et al. (2009). Learning multiple layers of features from tiny images.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105.

Li, Y., Wu, B., Feng, Y., Fan, Y., Jiang, Y., Li, Z., and Xia, S.-T. (2022). Semi-supervised robust training with generalized perturbed neighborhood. *Pattern Recognition*, 124:108472.

Liu, C., Salzmann, M., Lin, T., Tomioka, R., and Süsstrunk, S. (2020). On the loss landscape of adversarial training: Identifying challenges and how to overcome them. *arXiv preprint arXiv:2006.08403*.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. (2017). Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*.

Montasser, O., Hanneke, S., and Srebro, N. (2019). Vc classes are adversarially robustly learnable, but only improperly. In *Conference on Learning Theory*, pages 2512–2530. PMLR.

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

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. (2011). Reading digits in natural images with unsupervised feature learning.

Ozdaglar, A., Pattathil, S., Zhang, J., and Zhang, K. (2022). What is a good metric to study generalization of minimax learners? *arXiv preprint arXiv:2206.04502*.

Qin, Z., Fan, Y., Liu, Y., Zhang, Y., Wang, J., and Wu, B. (2022). Boosting the transferability of adversarial attacks with reverse adversarial perturbation.

Qin, Z., Fan, Y., Zha, H., and Wu, B. (2021). Random noise defense against query-based black-box attacks. *Advances in Neural Information Processing Systems*, 34:7650–7663.

Raghunathan, A., Xie, S. M., Yang, F., Duchi, J. C., and Liang, P. (2019). Adversarial training can hurt generalization. *arXiv preprint arXiv:1906.06032*.

Rebuffi, S.-A., Gowal, S., Calian, D. A., Stimberg, F., Wiles, O., and Mann, T. (2021). Fixing data augmentation to improve adversarial robustness. *arXiv preprint arXiv:2103.01946*.

Rice, L., Wong, E., and Kolter, Z. (2020). Overfitting in adversarially robust deep learning. In *International Conference on Machine Learning*, pages 8093–8104. PMLR.

Rogers, W. H. and Wagner, T. J. (1978). A finite sample distribution-free performance bound for local discrimination rules. *The Annals of Statistics*, pages 506–514.

Schmidt, L., Santurkar, S., Tsipras, D., Talwar, K., and Madry, A. (2018). Adversarially robust generalization requires more data. In *Advances in Neural Information Processing Systems*, pages 5014–5026.

Sinha, A., Namkoong, H., and Duchi, J. (2017). Certifiable distributional robustness with principled adversarial training. *arXiv preprint arXiv:1710.10571*, 2.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. (2013). Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Taheri, H., Pedarsani, R., and Thrampoulidis, C. (2020). Asymptotic behavior of adversarial training in binary classification. *arXiv preprint arXiv:2010.13275*.

Tramer, F., Carlini, N., Brendel, W., and Madry, A. (2020). On adaptive attacks to adversarial example defenses. *arXiv preprint arXiv:2002.08347*.

Vapnik, V. N. and Chervonenkis, A. Y. (2015). On the uniform convergence of relative frequencies of events to their probabilities. In *Measures of complexity*, pages 11–30. Springer.

Wang, Y., Ma, X., Bailey, J., Yi, J., Zhou, B., and Gu, Q. (2019). On the convergence and robustness of adversarial training. In *ICML*, volume 1, page 2.

Wong, E., Rice, L., and Kolter, J. Z. (2020). Fast is better than free: Revisiting adversarial training. *arXiv preprint arXiv:2001.03994*.

Wu, D., Xia, S.-T., and Wang, Y. (2020). Adversarial weight perturbation helps robust generalization. *arXiv preprint arXiv:2004.05884*.

Xiao, J., Fan, Y., Sun, R., and Luo, Z.-Q. (2022a). Adversarial rademacher complexity of deep neural networks.

Xiao, J., Qin, Z., Fan, Y., Wu, B., Wang, J., and Luo, Z.-Q. (2022b). Adaptive smoothness-weighted adversarial training for multiple perturbations with its stability analysis. *arXiv preprint arXiv:2210.00557*.

Xiao, J., Yang, L., Fan, Y., Wang, J., and Luo, Z.-Q. (2022c). Understanding adversarial robustness against on-manifold adversarial examples. *arXiv preprint arXiv:2210.00430*.

Xing, Y., Song, Q., and Cheng, G. (2021a). On the algorithmic stability of adversarial training. In *Thirty-Fifth Conference on Neural Information Processing Systems*.

Xing, Y., Song, Q., and Cheng, G. (2021b). On the generalization properties of adversarial training. In *International Conference on Artificial Intelligence and Statistics*, pages 505–513. PMLR.

Xing, Y., Zhang, R., and Cheng, G. (2021c). Adversarially robust estimate and risk analysis in linear regression. In *International Conference on Artificial Intelligence and Statistics*, pages 514–522. PMLR.

Yin, D., Kannan, R., and Bartlett, P. (2019). Rademacher complexity for adversarially robust generalization. In *International Conference on Machine Learning*, pages 7085–7094. PMLR.

Zhai, R., Cai, T., He, D., Dan, C., He, K., Hopcroft, J., and Wang, L. (2019). Adversarially robust generalization just requires more unlabeled data. *arXiv preprint arXiv:1906.00555*.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. (2021). Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115.

Zhang, J., Xiao, P., Sun, R., and Luo, Z. (2020). A single-loop smoothed gradient descent-ascent algorithm for nonconvex-concave min-max problems. *Advances in Neural Information Processing Systems*, 33:7377–7389.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [Yes]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [N/A]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [N/A]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Proof of the Theorem

A.1 Proof of Theorem 3.1

The proof can be found in (Hardt et al., 2016). We provide the proof in Appendix for reference. Denote by $S = (z_1, \ldots, z_n)$ and $S' = (z'_1, \ldots, z'_n)$ two independent random samples and let $S^{(i)} = (z_1, \ldots, z_{i-1}, z'_i, z_{i+1}, \ldots, z_n)$ be the sample that is identical to S except in the i'th example where we replace z_i with z'_i. With this notation, we get that

$$\begin{aligned}
\mathbb{E}_S \, \mathbb{E}_A \left[R_S[A(S)] \right] &= \mathbb{E}_S \, \mathbb{E}_A \left[\frac{1}{n} \sum_{i=1}^{n} h(A(S); z_i) \right] \\
&= \mathbb{E}_S \, \mathbb{E}_{S'} \, \mathbb{E}_A \left[\frac{1}{n} \sum_{i=1}^{n} h(A(S^{(i)}); z'_i) \right] \\
&= \mathbb{E}_S \, \mathbb{E}_{S'} \, \mathbb{E}_A \left[\frac{1}{n} \sum_{i=1}^{n} h(A(S); z'_i) \right] + \delta \\
&= \mathbb{E}_S \, \mathbb{E}_A \left[R_{\mathcal{D}}[A(S)] \right] + \delta,
\end{aligned}$$

where we can express δ as

$$\delta = \mathbb{E}_S \, \mathbb{E}_{S'} \, \mathbb{E}_A \left[\frac{1}{n} \sum_{i=1}^{n} h(A(S^{(i)}); z'_i) - \frac{1}{n} \sum_{i=1}^{n} h(A(S); z'_i) \right].$$

Furthermore, taking the supremum over any two data sets S, S' differing in only one sample, we can bound the difference as

$$|\delta| \leq \sup_{S, S', z} \mathbb{E}_A \left[h(A(S); z) - h(A(S'); z) \right] \leq \epsilon,$$

by our assumption on the uniform stability of A. The claim follows. □

A.2 Proof of Lemma 4.1

Let the adversarial examples for parameter θ_1 and θ_2 be

$$z_1 \in \arg \max_{\|z - z'\|_p \leq \epsilon} g(\theta_1, z')$$

$$z_2 \in \arg \max_{\|z - z'\|_p \leq \epsilon} g(\theta_2, z'),$$

then we have

$$\begin{aligned}
&\|h(\theta_1, z) - h(\theta_2, z)\| \\
=&|g(\theta_1, z_1) - g(\theta_2, z_2)| \\
\leq& \max\{|g(\theta_1, z_1) - g(\theta_2, z_1)|, |g(\theta_1, z_2) - g(\theta_2, z_2)|\} \\
\leq& L\|\theta_1 - \theta_2\|,
\end{aligned}$$

where the first inequality is based on the fact that $g(\theta_1, z_1) \geq g(\theta_1, z_2)$ and $g(\theta_2, z_2) \geq g(\theta_2, z_1)$, the second inequality is based on Assumption 4.1. This proves Lemma 4.1.1.

For all subgradient $d(\theta, z) \in \partial_\theta h(\theta, z)$, we have

$$\begin{aligned}
&\|d(\theta_1, z) - d(\theta_2, z)\| \\
=&\|\nabla_\theta g(\theta_1, z_1) - \nabla_\theta g(\theta_2, z_2)\| \\
\leq&\|\nabla_\theta g(\theta_1, z_1) - \nabla_\theta g(\theta_2, z_1)\| + \|\nabla_\theta g(\theta_2, z_1) - \nabla_\theta g(\theta_2, z_2)\| \\
\leq& L_\theta \|\theta_1 - \theta_2\| + L_z \|z_1 - z_2\|_p \\
\leq& L_\theta \|\theta_1 - \theta_2\| + L_z [\|z_1 - z\|_p + \|z - z_2\|_p] \\
\leq& L_\theta \|\theta_1 - \theta_2\| + 2L_z \epsilon
\end{aligned}$$

where the first and the third inequality is due to triangle inequality, the second inequality is based on Assumption 4.1. This proves the second inequality (non-gradient Lipschitz) in Lemma 4.1. □

A.3 Proof of Lemma 4.2

Proof of Lemma 4.2.1 (η-approximate descent Lemma).
Let $\tilde{\theta}$ be a point in the line segment of θ_1 and θ_2, $\tilde{\theta}(u) = \theta_2 + u(\theta_1 - \theta_2)$, then

$$\begin{aligned}
&h(\theta_1) - h(\theta_2) \\
=&\int_0^1 \langle \theta_1 - \theta_2, \nabla_\theta h(\tilde{\theta}(u))\rangle du \\
=&\int_0^1 \langle \theta_1 - \theta_2, \nabla_\theta h(\theta_2) + \nabla_\theta h(\tilde{\theta}(u)) - \nabla_\theta h(\theta_2)\rangle du \\
=&\langle \nabla_\theta h(\theta_2), \theta_1 - \theta_2\rangle + \int_0^1 \langle \theta_1 - \theta_2, \nabla_\theta h(\tilde{\theta}(u)) - \nabla_\theta h(\theta_2)\rangle du \\
\leq&\langle \nabla_\theta h(\theta_2), \theta_1 - \theta_2\rangle + \int_0^1 \|\theta_1 - \theta_2\|\|\nabla_\theta h(\tilde{\theta}(u)) - \nabla_\theta h(\theta_2)\| du \\
\leq&\langle \nabla_\theta h(\theta_2), \theta_1 - \theta_2\rangle + \int_0^1 \|\theta_1 - \theta_2\|[\beta\|\tilde{\theta}(u) - \theta_2\| + \eta] du \\
=&\langle \nabla_\theta h(\theta_2), \theta_1 - \theta_2\rangle + \int_0^1 \|\theta_1 - \theta_2\|[\beta u\|\theta_1 - \theta_2\| + \eta] du \\
=&\langle \nabla_\theta h(\theta_2), \theta_1 - \theta_2\rangle + \beta\|\theta_1 - \theta_2\|^2 \int_0^1 u du + \eta\|\theta_1 - \theta_2\|] \\
=&\langle \nabla_\theta h(\theta_2), \theta_1 - \theta_2\rangle + \frac{\beta}{2}\|\theta_1 - \theta_2\|^2 + \eta\|\theta_1 - \theta_2\|.
\end{aligned}$$

□

Proof of Lemma 4.2.2 (η-approximate co-coercive).
By Lemma 4.2.1 (η-approximate descent Lemma), we have

$$h(\theta_1) \leq h(\theta_2) + \langle \nabla_\theta h(\theta_2), \theta_1 - \theta_2\rangle + \frac{\beta}{2}\|\theta_1 - \theta_2\|^2 + \eta\|\theta_1 - \theta_2\|.$$

Let θ^* be a minimizer of h, then

$$\begin{aligned}
h(\theta^*) = \inf_{\theta_1} h(\theta_1) \leq \inf_{\theta_1} &\left(h(\theta_2) + \langle \nabla_\theta h(\theta_2), \theta_1 - \theta_2\rangle + \frac{\beta}{2}\|\theta_1 - \theta_2\|^2 + \eta\|\theta_1 - \theta_2\| \right) \\
= \inf_{\|v\|=1} \inf_{t\geq 0} &\left(h(\theta_2) + t\nabla_\theta h(\theta_2)^T v + \frac{\beta t^2}{2} + \eta t \right),
\end{aligned}$$

where $t = \|\theta_1 - \theta_2\|$ and $v = (\theta_1 - \theta_2)/\|\theta_1 - \theta_2\|$. Then

$$\begin{aligned}
&\inf_{\|v\|=1} \inf_{t\geq 0} \left(h(\theta_2) + t\nabla_\theta h(\theta_2)^T v + \frac{\beta t^2}{2} + \eta t \right) \\
=&\inf_{t\geq 0} \left(h(\theta_2) - t\|\nabla_\theta h(\theta_2)\| + \frac{\beta t^2}{2} + \eta t \right) \\
=&h(\theta_2) + \inf_{t\geq 0} \left(-t(\|\nabla_\theta h(\theta_2)\| - \eta) + \frac{\beta t^2}{2} \right).
\end{aligned}$$

If $\|\nabla_\theta h(\theta_2)\| - \eta \leq 0$, the quaduatic function is optimized when $t = 0$. Then

$$\begin{aligned}
&h(\theta_2) + \inf_{t\geq 0} \left(-t(\|\nabla_\theta h(\theta_2)\| - \eta) + \frac{\beta t^2}{2} \right) \\
=&h(\theta_2).
\end{aligned}$$

If $\|\nabla_\theta h(\theta_2)\| - \eta \geq 0$, then

$$\begin{aligned}
&h(\theta_2) + \inf_{t\geq 0} \left(-t(\|\nabla_\theta h(\theta_2)\| - \eta) + \frac{\beta t^2}{2} \right) \\
=&h(\theta_2) - \frac{1}{2\beta}[\|\nabla_\theta h(\theta_2)\| - \eta]^2.
\end{aligned}$$

Therefore, we obtain that

$$h(\theta^*) - h(\theta) \leq -\frac{1}{2\beta}\left[[\|\nabla h(\theta)\| - \eta]_+\right]^2. \tag{A.1}$$

Define

$$h_1(w) = h(w) - \nabla h(\theta_1)^T w$$

and

$$h_2(w) = h(w) - \nabla h(\theta_2)^T w.$$

Firstly, it is easy to see that $h_1(w)$ and $h_2(w)$ are both η-approximate β-gradient Lipschitz, which satisfies inequatily in Eq. (A.1). Secondly, $w = \theta_1$ minimizes $h_1(w)$. Then

$$\begin{aligned}
&h(\theta_2) - h(\theta_1) - \nabla h(\theta_1)^T(\theta_2 - \theta_1)\\
=&h_1(\theta_2) - h_1(\theta_1)\\
\geq&\frac{1}{2\beta}\left[[\|\nabla h_1(\theta_2)\| - \eta]_+\right]^2\\
=&\frac{1}{2\beta}\left[[\|\nabla h(\theta_1) - \nabla h(\theta_2)\| - \eta]_+\right]^2.
\end{aligned} \tag{A.2}$$

Similarly, we have

$$h(\theta_1) - h(\theta_2) - \nabla h(\theta_2)^T(\theta_1 - \theta_2) \geq \frac{1}{2\beta}\left[[\|\nabla h(\theta_1) - \nabla h(\theta_2)\| - \eta]_+\right]^2. \tag{A.3}$$

Take the summation of Eq. (A.2) and Eq. (A.3), we have

$$\langle \nabla h(\theta_1) - \nabla h(\theta_2), \theta_1 - \theta_2\rangle \geq \frac{1}{\beta}\left[[\|\nabla h(\theta_1) - \nabla h(\theta_2)\| - \eta]_+\right]^2.$$

□

A.4 Proof of Lemma 4.3

Proof of Lemma 4.3.1 ($\alpha\eta$-approximately $(1 + \alpha\beta)$-expansive).

$$\begin{aligned}
&\|G_{\alpha,z}(\theta_1) - G_{\alpha,z}(\theta_2)\|\\
=&\|\theta_1 - \theta_2 - \alpha(\nabla h(\theta_1) - \nabla h(\theta_2))\|\\
\leq&\|\theta_1 - \theta_2\| + \|\alpha(\nabla h(\theta_1) - \nabla h(\theta_2))\|\\
\leq&\|\theta_1 - \theta_2\| + \alpha(\beta\|\theta_1 - \theta_2\| + \eta)\\
\leq&(1 + \alpha\beta)\|\theta_1 - \theta_2\| + \alpha\eta.
\end{aligned}$$

□

Proof of Lemma 4.3.2 ($\alpha\eta$-approximately non-expansive.) Let t = $\|\nabla h(\theta_1) - \nabla h(\theta_2)\|$.
If $t \leq \eta$, we have

$$\begin{aligned}
&\|G_{\alpha,z}(\theta_1) - G_{\alpha,z}(\theta_2)\|\\
=&\|\theta_1 - \theta_2 - \alpha(\nabla h(\theta_1) - \nabla h(\theta_2))\|\\
\leq&\|\theta_1 - \theta_2\| + \alpha\|\nabla h(\theta_1) - \nabla h(\theta_2)\|\\
\leq&\|\theta_1 - \theta_2\| + \alpha\eta.
\end{aligned}$$

If $t \geq \eta$, we have

$$
\begin{aligned}
&\|G_{\alpha,z}(\theta_1) - G_{\alpha,z}(\theta_2)\|^2 \\
=&\|\theta_1 - \theta_2 - \alpha(\nabla h(\theta_1) - \nabla h(\theta_2))\|^2 \\
=&\|\theta_1 - \theta_2\|^2 - 2\alpha(\nabla h(\theta_1) - \nabla h(\theta_2)^T(\theta_1 - \theta_2) + \alpha^2 t^2 \\
\leq&\|\theta_1 - \theta_2\|^2 - \frac{2\alpha}{\beta}(t-\eta)^2 + \alpha^2 t^2 \\
=&\|\theta_1 - \theta_2\|^2 - \frac{2\alpha}{\beta}(t-\eta)^2 - \frac{2\alpha\eta}{\beta}(t-\eta) + \alpha^2 t^2 + \frac{2\alpha\eta}{\beta}(t-\eta) \\
=&\|\theta_1 - \theta_2\|^2 - \frac{2\alpha t}{\beta}(t-\eta) + \alpha^2 t^2 + \frac{2\alpha\eta}{\beta}(t-\eta).
\end{aligned}
$$

Let $\alpha \leq 1/\beta$, then

$$
\begin{aligned}
&\|\theta_1 - \theta_2\|^2 - \frac{2\alpha t}{\beta}(t-\eta) + \alpha^2 t^2 + \frac{2\alpha\eta}{\beta}(t-\eta) \\
\leq&\|\theta_1 - \theta_2\|^2 - 2\alpha^2 t(t-\eta) + \alpha^2 t^2 + \frac{2\alpha\eta}{\beta}(t-\eta) \\
\leq&\|\theta_1 - \theta_2\|^2 - \alpha^2(t+\eta)(t-\eta) + \alpha^2 t^2 + \frac{2\alpha\eta}{\beta}(t-\eta) \\
\leq&\|\theta_1 - \theta_2\|^2 + \alpha^2\eta^2 + \frac{2\alpha\eta}{\beta}(t-\eta).
\end{aligned}
$$

By the definition of η-approximate smoothness,

$$
\frac{1}{\beta}(t-\eta) \leq \|\theta_1 - \theta_2\|.
$$

Then

$$
\begin{aligned}
&\|\theta_1 - \theta_2\|^2 + \alpha^2\eta^2 + \frac{2\alpha\eta}{\beta}(t-\eta) \\
\leq&\|\theta_1 - \theta_2\|^2 + \alpha^2\eta^2 + 2\alpha\eta\|\theta_1 - \theta_2\| \\
=&(\|\theta_1 - \theta_2\| + \alpha\eta)^2.
\end{aligned}
$$

Therefore, we obtain that

$$
\|G_{\alpha,z}(\theta_1) - G_{\alpha,z}(\theta_2)\| \leq \|\theta_1 - \theta_2\| + \alpha\eta.
$$

□

Proof of Lemma 4.3.3 ($\alpha\eta$-approximately $(1-\alpha\gamma)$-contraction.).
Firstly, if $h(\theta)$ is a γ-strongly convex, η-approximately β-gradient Lipschitz function, $\phi(\theta) = h(\theta) - \frac{\gamma}{2}\|\theta\|^2$ is a convex, η-approximate $(\beta-\gamma)$-gradient Lipschitz function. The proof of convexity follows the definition. To see the second claim, since

$$
\begin{aligned}
&\phi(\theta_1) - \phi(\theta_2) \\
=&h(\theta_1) - h(\theta_2) - (\frac{\gamma}{2}\|\theta_1\|^2 - \frac{\gamma}{2}\|\theta_2\|^2) \\
\leq&\langle\nabla_\theta h(\theta_2), \theta_1 - \theta_2\rangle + \frac{\beta}{2}\|\theta_1 - \theta_2\|^2 + \eta\|\theta_1 - \theta_2\| - (\frac{\gamma}{2}\|\theta_1\|^2 - \frac{\gamma}{2}\|\theta_2\|^2) \\
\leq&\langle\nabla_\theta \phi(\theta_2), \theta_1 - \theta_2\rangle + \frac{\beta}{2}\|\theta_1 - \theta_2\|^2 + \eta\|\theta_1 - \theta_2\| - (\frac{\gamma}{2}\|\theta_1\|^2 - \frac{\gamma}{2}\|\theta_2\|^2) + \gamma\theta_2^T(\theta_1 - \theta_2) \\
\leq&\langle\nabla_\theta \phi(\theta_2), \theta_1 - \theta_2\rangle + \frac{\beta}{2}\|\theta_1 - \theta_2\|^2 + \eta\|\theta_1 - \theta_2\| - \frac{\gamma}{2}\|\theta_1 - \theta_2\|^2 \\
\leq&\langle\nabla_\theta \phi(\theta_2), \theta_1 - \theta_2\rangle + \frac{\beta-\gamma}{2}\|\theta_1 - \theta_2\|^2 + \eta\|\theta_1 - \theta_2\|.
\end{aligned}
$$

Therefore, $\phi(\theta)$ satisfies the η-approximate $(\beta-\gamma)$-descent Lemma. Let $t = \|\nabla\phi(\theta_1) - \nabla\phi(\theta_2)\|$. If $t \le \eta$, we have

$$\begin{aligned}
&\|G_{\alpha,z}(\theta_1) - G_{\alpha,z}(\theta_2)\| \\
=&\|\theta_1 - \theta_2 - \alpha(\nabla h(\theta_1) - \nabla h(\theta_2))\| \\
=&\|\theta_1 - \theta_2 - \alpha(\nabla\phi(\theta_1) + \gamma\theta_1 - \nabla\phi(\theta_2) - \gamma\theta_2)\| \\
\le&\|(1-\alpha\gamma)(\theta_1 - \theta_2)\| + \alpha\|\nabla\phi(\theta_1) - \nabla\phi(\theta_2)\| \\
\le&(1-\alpha\gamma)\|\theta_1 - \theta_2\| + \alpha\eta.
\end{aligned}$$

If $t \ge \eta$, we have

$$\begin{aligned}
&\|G_{\alpha,z}(\theta_1) - G_{\alpha,z}(\theta_2)\|^2 \\
=&\|\theta_1 - \theta_2 - \alpha(\nabla h(\theta_1) - \nabla h(\theta_2))\|^2 \\
=&\|\theta_1 - \theta_2 - \alpha(\nabla\phi(\theta_1) + \gamma\theta_1 - \nabla\phi(\theta_2) - \gamma\theta_2)\|^2 \\
\le&(1-\alpha\gamma)^2\|\theta_1 - \theta_2\|^2 - 2\alpha(1-\alpha\gamma)(\nabla\phi(\theta_1) - \nabla\phi(\theta_2)^T(\theta_1 - \theta_2) + \alpha^2 t^2 \\
\le&(1-\alpha\gamma)^2\|\theta_1 - \theta_2\|^2 - \frac{2\alpha(1-\alpha\gamma)}{\beta-\gamma}(t-\eta)^2 + \alpha^2 t^2 \\
\le&(1-\alpha\gamma)^2\|\theta_1 - \theta_2\|^2 - \frac{2\alpha(1-\alpha\gamma)}{\beta-\gamma}(t-\eta)^2 - \frac{2\alpha(1-\alpha\gamma)\eta}{\beta-\gamma}(t-\eta) + \alpha^2 t^2 + \frac{2\alpha(1-\alpha\gamma)\eta}{\beta-\gamma}(t-\eta).
\end{aligned}$$

Since $\alpha \le 1/\beta$, we have $(1-\alpha\gamma)/(\beta-\gamma) \ge \alpha$, then

$$\begin{aligned}
&(1-\alpha\gamma)^2\|\theta_1 - \theta_2\|^2 - \frac{2\alpha(1-\alpha\gamma)}{\beta-\gamma}(t-\eta)^2 - \frac{2\alpha(1-\alpha\gamma)\eta}{\beta-\gamma}(t-\eta) + \alpha^2 t^2 + \frac{2\alpha(1-\alpha\gamma)\eta}{\beta-\gamma}(t-\eta) \\
\le&(1-\alpha\gamma)^2\|\theta_1 - \theta_2\|^2 - \alpha^2 t(t-\eta) + \alpha^2 t^2 + \frac{2\alpha(1-\alpha\gamma)\eta}{\beta-\gamma}(t-\eta) \\
\le&(1-\alpha\gamma)^2\|\theta_1 - \theta_2\|^2 + \alpha^2\eta^2 + 2\alpha(1-\alpha\gamma)\eta\|\theta_1 - \theta_2\| \\
\le&\Big((1-\alpha\gamma)\|\theta_1 - \theta_2\| + \alpha\eta\Big)^2.
\end{aligned}$$

Therefore, we obtain that

$$\|G_{\alpha,z}(\theta_1) - G_{\alpha,z}(\theta_2)\| \le (1-\alpha\gamma)\|\theta_1 - \theta_2\| + \alpha\eta.$$

□

A.5 Proof of Theorem 5.1

The proof follows the standard techniques for uniform stability. We need to replace the non-expansive property used in standard analysis by the approximately non-expansive property. Let S and S' be two samples of size n differing in only a single example. Consider two trajectories $\theta_1^1, \ldots, \theta_1^T$ and $\theta_2^1, \ldots, \theta_2^T$ induced by running SGD on sample S and S', respectively. Let $\delta_t = \|\theta_1^t - \theta_2^t\|$.

Fixing an example $z \in Z$ and apply the Lipschitz condition on $h(\cdot\,; z)$, we have

$$\mathbb{E}\left|h(\theta_1^T; z) - h(\theta_2^T; z)\right| \le L\,\mathbb{E}\left[\delta_T\right]. \tag{A.4}$$

Observe that at step t, with probability $1 - 1/n$, the example selected by SGD is the same in both S and S'. With probability $1/n$ the selected example is different. Therefore, by the $\alpha\eta$-approximate non-expansive property, we have

$$\mathbb{E}\left[\delta_{t+1}\right] \le \left(1 - \frac{1}{n}\right)\left(\mathbb{E}\left[\delta_t\right] + \alpha_t\eta\right) + \frac{1}{n}\,\mathbb{E}\left[\delta_t\right] + \frac{2\alpha_t L}{n} \le \mathbb{E}\left[\delta_t\right] + \left(\eta + \frac{2L}{n}\right)\alpha_t\,. \tag{A.5}$$

Unraveling the recursion gives

$$\mathbb{E}\left[\delta_T\right] \le \left(\eta + \frac{2L}{n}\right)\sum_{t=1}^{T}\alpha_t\,.$$

Plugging this back into Eq. (A.4), we obtain

$$\mathcal{E}_{gen} \leq L\left(\eta + \frac{2L}{n}\right)\sum_{t=1}^{T}\alpha_t\,.$$

Since this bounds holds for all S, S' and z, we obtain the desired bound on the uniform stability. □

A.6 Proof of Theorem 5.2

Proof: The construction of function h is adopted from the construction in the work of (Bassily et al., 2020).
Let $T \leq d$, and $v, K \geq 0$. Considering $\mathcal{Z} = \{0, 1\}$, and the objective function

$$h(\theta, z) = \begin{cases} \eta \max\{0, x_1 - v, \cdots, x_T - v\} & \text{if } z = 0 \\ \langle r, x\rangle / K & \text{if } z = 1, \end{cases} \tag{A.6}$$

where $r = (-1, \cdots, -1, 0, \cdots, 0)$ (i.e., equals to -1 for the first T components). Function h is η-approximately smooth since the first case is a piece-wise linear function. For the dataset S and S' differ in at most one sample, the empirical objective functions are

$$R_S(\theta) = \frac{1}{nK}\langle r, x\rangle + \frac{n-1}{n}\eta \max\{0, x_1 - v, \cdots, x_T - v\},$$

and

$$R_{S'}(\theta) = \eta \max\{0, x_1 - v, \cdots, x_T - v\}.$$

Let θ_1^T and θ_2^T be the trajectories running algorithm on dataset S and S', initialized on $\theta_1^0 = \theta_2^0 = 0$. Clearly, $\theta_2^t = 0$ for all t. It is easy to obtain $\theta_1^t = -\frac{t\alpha r}{nk} - \alpha\eta\frac{n-1}{n}\sum_{s=1}^{t-1} e_s$ recursively. By the orthogonality of the subgradients, we have

$$\delta(S, S') = \|\theta_1^T - \theta_2^T\| = \|\theta_1^T\| \geq \Omega\left(\alpha\eta\left\|\sum_{s=1}^{T} e_s\right\|\right) = \Omega(\alpha\eta\sqrt{T}). \tag{A.7}$$

On the other hand, the work of (Hardt et al., 2016) provided a lower bound for general non-smooth function

$$\delta(S, S') \geq \Omega\left(\frac{L\alpha T}{n}\right). \tag{A.8}$$

Combining Eq. (A.7) and Eq. (A.8), we have

$$\delta(S, S') \geq \Omega\left(\alpha\eta\sqrt{T} + \frac{L\alpha T}{n}\right). \tag{A.9}$$

□

A.7 Proof of Theorem 5.3

We consider a general form of Theorem 5.3.

Theorem A.1 (Non-convex)**.** *Assume that $h(\theta, z)$ is L-Lipschitz, and η-approximately β-gradient Lipschitz in θ for all given $z \in \mathcal{Z}$. Assume in addition that $0 \leq g(\theta, z) \leq B$ for all θ and z. Suppose that we run SGD on the adversarial surrogate loss with step sizes $\alpha_t \leq c/t$ for T steps, where $c > 0$. Then, for all $t_0 \in \{1, 2, \cdots, n\}$, adversarial training satisfies uniform stability with*

$$\mathcal{E}_{gen} = \mathbb{E}[R_{\mathcal{D}}(\theta^T) - R_S(\theta^T)] \leq \frac{Bt_0}{n-1} + \frac{1}{\beta(n-1)}\left(2L^2 + L\eta n\right)\left(\frac{T}{t_0}\right)^{\beta c}. \tag{A.10}$$

Let $q = \beta c$. For small T (s.t. $t_0 \leq n$ when we set the first term equals to the second term). we select t_0 to optimize the right hand side, then

$$\mathcal{E}_{gen} \leq \frac{2}{n-1}B^{\frac{q}{q+1}}\left(\frac{2L^2 + L\eta n}{\beta}\right)^{\frac{1}{q+1}} T^{\frac{q}{q+1}}. \tag{A.11}$$

For arbitrary T, the optimal $t_0 > n$. We simply let $t_0 = 1$, then

$$\mathcal{E}_{gen} \leq \frac{BL_\theta + (2L^2 + L\eta n)T^q}{\beta(n-1)}. \tag{A.12}$$

Proof: Let S and S' be two samples of size n differing in only a single example. Consider two trajectories $\theta_1^1, \ldots, \theta_1^T$ and $\theta_2^1, \ldots, \theta_2^T$ induced by running SGD on sample S and S', respectively. Let $\delta_t = \|\theta_1^t - \theta_2^t\|$. Let $t_0 \in \{0, 1, \ldots, n\}$, be the iteration that $\delta_{t_0} = 0$, but SGD picks two different samples form S and S' in iteration $t_0 + 1$, then

$$\mathcal{E}_{gen} \leq \frac{t_0}{n} B + L \, \mathbb{E}\left[\delta_T \mid \delta_{t_0} = 0\right] . \tag{A.13}$$

Let $\Delta_t = \mathbb{E}\left[\delta_t \mid \delta_{t_0} = 0\right]$. Observe that at step t, with probability $1 - 1/n$, the example selected by SGD is the same in both S and S'. With probability $1/n$ the selected example is different. Therefore, by the $\alpha\eta$-approximate $(1 + \alpha\beta)$-expansive property, for every $t \geq t_0$,

$$\begin{aligned}
\Delta_{t+1} &\leq \left(1 - \frac{1}{n}\right)(1 + \alpha_t \beta)\Delta_t + \frac{1}{n}\Delta_t + \left(\eta + \frac{2L}{n}\right)\alpha_t \\
&\leq \left(\frac{1}{n} + (1 - 1/n)(1 + c\beta/t)\right)\Delta_t + \left(\eta + \frac{2L}{n}\right)\frac{c}{t} \\
&= \left(1 + (1 - 1/n)\frac{c\beta}{t}\right)\Delta_t + \left(\eta + \frac{2L}{n}\right)\frac{c}{t} \\
&\leq \exp\left((1 - 1/n)\frac{c\beta}{t}\right)\Delta_t + \left(\eta + \frac{2L}{n}\right)\frac{c}{t} .
\end{aligned}$$

Here we used the fact that $1 + x \leq \exp(x)$ for all x.

Using the fact that $\Delta_{t_0} = 0$, we can unwind this recurrence relation from T down to $t_0 + 1$. This gives

$$\begin{aligned}
\Delta_T &\leq \sum_{t=t_0+1}^{T} \left\{ \prod_{k=t+1}^{T} \exp\left((1 - \tfrac{1}{n})\tfrac{\beta c}{k}\right) \right\} \left(\eta + \frac{2L}{n}\right)\frac{c}{t} \\
&= \sum_{t=t_0+1}^{T} \exp\left((1 - \tfrac{1}{n})\beta c \sum_{k=t+1}^{T} \tfrac{1}{k}\right) \left(\eta + \frac{2L}{n}\right)\frac{c}{t} \\
&\leq \sum_{t=t_0+1}^{T} \exp\left((1 - \tfrac{1}{n})\beta c \log(\tfrac{T}{t})\right) \left(\eta + \frac{2L}{n}\right)\frac{c}{t} \\
&= \left(\eta + \frac{2L}{n}\right) c T^{\beta c(1-1/n)} \sum_{t=t_0+1}^{T} t^{-\beta c(1-1/n)-1} \\
&\leq \left(\eta + \frac{2L}{n}\right) \frac{1}{(1 - 1/n)\beta c} c \left(\frac{T}{t_0}\right)^{\beta c(1-1/n)} \\
&\leq \frac{\eta n + 2L}{\beta(n-1)} \left(\frac{T}{t_0}\right)^{\beta c} ,
\end{aligned}$$

Plugging this bound into (A.13), we get

$$\mathcal{E}_{gen} \leq \frac{B t_0}{n-1} + \frac{L\eta n + 2L^2}{\beta(n-1)} \left(\frac{T}{t_0}\right)^{\beta c} .$$

Let $q = \beta c$. For small T (s.t. $t_0 \leq n$ when we set the first term equals to the second term). we select t_0 to optimize the right hand side, then

$$\mathcal{E}_{gen} \leq \frac{2}{n-1} B^{\frac{q}{q+1}} \left(\frac{2L^2 + L\eta n}{\beta}\right)^{\frac{1}{q+1}} T^{\frac{q}{q+1}} .$$

For arbitrary T, the optimal $t_0 > n$. We simply let $t_0 = 1$, then

$$\mathcal{E}_{gen} \leq \frac{B L_\theta + (2L^2 + L\eta n) T^q}{\beta(n-1)} .$$

Since the bound we just derived holds for all S, S' and z, we immediately get the claimed upper bound on the uniform stability. Let $q = 1$, we obtain the result of Theorem 5.3. □

A.8 Proof of Theorem 5.4

The proof follows the idea in convex case. By the $\alpha\eta$-approximately $(1-\alpha\gamma)$-contraction, for every t,

$$\begin{aligned}\mathbb{E}\,\delta_{t+1} &\leq \left(1-\frac{1}{n}\right)(1-\alpha\gamma)\,\mathbb{E}\,\delta_t + \frac{1}{n}(1-\alpha\gamma)\,\mathbb{E}\,\delta_t + \left(\eta+\frac{2L}{n}\right)\alpha \\ &= (1-\alpha\gamma)\,\mathbb{E}\,\delta_t + \left(\eta+\frac{2L}{n}\right)\alpha\,.\end{aligned} \tag{A.14}$$

Unraveling the recursion gives

$$\mathbb{E}\,\delta_T \leq \left(\eta+\frac{2L}{n}\right)\alpha\sum_{t=0}^{T}(1-\alpha\gamma)^t \leq \frac{\eta}{\gamma}+\frac{2L}{\gamma n}\,.$$

Plugging the above inequality into Eq. (A.4), we obtain

$$\mathcal{E}_{gen} \leq \frac{L\eta}{\gamma}+\frac{2L^2}{\gamma n}\,.$$

Since this bounds holds for all S, S' and z, the Theorem follows. □

A.9 Proof of Theorem 6.2

Let $\bar{\theta} = \frac{1}{T}\sum_{t=1}^{T}\theta^t$ denote the average of the stochastic gradient iterates. Since

$$\theta^t = \sum_{k=1}^{t}\alpha_k\nabla h(\theta^k; z_k)\,,$$

we have

$$\bar{\theta} = \sum_{t=1}^{T}\alpha_t\frac{T-t+1}{T}\nabla h(\theta^k; z_k)$$

Using the $\alpha\eta$-approximate non-expansive, we have

$$\delta_t \leq (1-1/n)\delta_{t-1} + \frac{1}{n}\left(\delta_{t-1}+(\eta n+2L)\alpha_t\frac{T-t+1}{T}\right)\,.$$

which implies

$$\delta_T \leq \left(\eta+\frac{2L}{n}\right)\sum_{t=1}^{T}\alpha_t\frac{T-t+1}{T} = \left(\frac{\eta}{2}+\frac{L}{n}\right)\sum_{t=1}^{T}\alpha_t\,.$$

Since f is L-Lipschitz, we have

$$\mathcal{E}_{gen}(\bar{\theta}) \leq \left(\frac{L\eta}{2}+\frac{L^2}{n}\right)\sum_{t=1}^{T}\alpha_t. \tag{A.15}$$

Here the expectation is taken over the algorithm and hence the claim follows by our definition of uniform stability. $\mathcal{E}_{opt}$ follows (Nemirovski et al., 2009). □

B Discussion on Non-convex and Strongly Convex Case

B.1 Discussion on Non-convex Case

To discuss the generalization-optimization trade-off in the non-convex case. We first give the optimization error bound.

Theorem B.1. *Assume that h is η-approximate β-gradient Lipschitz and given $0<\tau<1$. Without loss of generality, assume the stochastic gradient $\nabla\tilde{h}(\theta)$ be unbiased and have a bounded variance σ^2. Let the stochastic gradient descent (SGD) update be $\theta_{t+1}=\theta_t-\alpha\nabla\tilde{h}(\theta_t)$ with a constant step size $\alpha = 1/\sqrt{T}$ for number of iterations $T \geq (\beta/2(1-\tau))^2$. $\exists t \leq T$, s.t.*

$$\mathbb{E}\|\nabla h(\theta_t)\|^2 \leq \frac{\eta^2}{\tau^2}+\frac{2\eta\sigma}{\tau}+\mathcal{O}(\frac{1}{\sqrt{T}}). \tag{B.1}$$

Proof:

Assume that the stochastic gradient $\nabla h(\theta)$ be unbiased and have a bounded variance σ^2.

$$\mathbb{E}[\nabla \tilde{h}(\theta)] = \nabla h(\theta),$$
$$\mathbb{E}\|\nabla \tilde{h}(\theta)\|^2 \leq \|\nabla h(\theta)\|^2 + \sigma^2.$$

Notice that when θ is a random vector, the above expectation is condition on θ. Let the stochastic gradient descent (SGD) update be $\theta_{t+1} = \theta_t - \alpha \tilde{h}(\theta_t)$ with a constant step size $\alpha = 1/\sqrt{T}$. By η-approximately Descent Lemma, we have

$$\begin{aligned}
&h(\theta_{t+1}) - h(\theta_t) \\
=& -\alpha \langle \nabla h(\theta_t), \theta_{t+1} - \theta_t \rangle + \frac{\beta}{2}\|\theta_{t+1} - \theta_t\|^2 + \eta \|\theta_{t+1} - \theta_t\| \\
=& -\alpha \langle \nabla h(\theta_t), \nabla \tilde{h}(\theta_t) \rangle + \frac{\beta \alpha^2}{2}\|\nabla \tilde{h}(\theta_t)\|^2 + \eta\alpha \|\nabla \tilde{h}(\theta_t)\|.
\end{aligned}$$

Given θ_t, take the conditional expectation over the noised introduced by SGD, we have

$$\begin{aligned}
&\mathbb{E}[h(\theta_{t+1})] - h(\theta_t) \\
=& -\alpha \langle \nabla h(\theta_t), \mathbb{E}[\nabla \tilde{h}(\theta_t)] \rangle + \frac{\beta \alpha^2}{2}\mathbb{E}\|\nabla \tilde{h}(\theta_t)\|^2 + \eta\alpha \mathbb{E}\|\nabla \tilde{h}(\theta_t)\| \\
\leq& -\alpha \|\nabla h(\theta_t)\|^2 + \frac{\beta \alpha^2}{2}\left[\|\nabla h(\theta_t)\|^2 + \sigma^2\right] + \eta\alpha \sqrt{\left[\mathbb{E}\|\nabla \tilde{h}(\theta_t)\|\right]^2} \\
\leq& -\alpha \|\nabla h(\theta_t)\|^2 + \frac{\beta \alpha^2}{2}\left[\|\nabla h(\theta_t)\|^2 + \sigma^2\right] + \eta\alpha \sqrt{\|\nabla h(\theta_t)\|^2 + \sigma^2} \\
\leq& -\alpha \|\nabla h(\theta_t)\|^2 + \frac{\beta \alpha^2}{2}\left[\|\nabla h(\theta_t)\|^2 + \sigma^2\right] + \eta\alpha \left[\|\nabla h(\theta_t)\| + \sigma\right] \\
=& -\alpha \|\nabla h(\theta_t)\|^2 + \frac{\beta \alpha^2}{2}\|\nabla h(\theta_t)\|^2 + \eta\alpha \|\nabla_\theta h(\theta_t)\| + \frac{\beta \alpha^2 \sigma^2}{2} + \eta\alpha\sigma \\
\leq& -\tau\alpha \|\nabla h(\theta_t)\|^2 + \eta\alpha \|\nabla h(\theta_t)\| + \frac{\beta \alpha^2 \sigma^2}{2} + \eta\alpha\sigma,
\end{aligned}$$

where the first inequality is the assumption of SGD, the second inequality is the Jensen's inequality, the third one is the assumption of SGD, the fourth one is the Cauchy-Schwartz inequality, and the last one is because of the size of step size α. Take the expectation over the trajectory $\theta_0, \theta_1, \cdots, \theta_T$, and take the average aver $t = 0, 1, \cdots, T$, we have

$$\begin{aligned}
&\frac{1}{T}\sum_{t=0}^{T}\left[\tau\alpha \mathbb{E}\|\nabla h(\theta_t)\|^2 - \eta\alpha \mathbb{E}\|\nabla h(\theta_t)\|\right] \\
\leq& \frac{1}{T}\sum_{t=0}^{T}\left[\mathbb{E}[h(\theta_t)] - \mathbb{E}h(\theta_{t+1})\right] + \frac{\beta \alpha^2 \sigma^2}{2} + \eta\alpha\sigma \\
\leq& \frac{1}{T}\left[\mathbb{E}[h(\theta_0)] - h(\theta_*)\right] + \frac{\eta \alpha^2 \sigma^2}{2} + \eta\alpha\sigma.
\end{aligned}$$

Let $\mathbb{E}[h(\theta_0)] - h(\theta_*) = D$ and divide α on both side. $\exists t \leq T$, $s.t.$

$$\tau \mathbb{E}\|\nabla h(\theta_t)\|^2 - \eta \mathbb{E}\|\nabla h(\theta_t)\| \leq \frac{D}{T\alpha} + \frac{\beta\alpha\sigma^2}{2} + \eta\sigma.$$

Since $\alpha = 1/\sqrt{T}$, we have

$$\begin{aligned}
&\tau \mathbb{E}\|\nabla h(\theta_t)\|^2 - \eta \mathbb{E}\|\nabla h(\theta_t)\| \leq \frac{D}{\sqrt{T}} + \frac{\beta\sigma^2}{2\sqrt{T}} + \eta\sigma \\
\Leftrightarrow& \mathbb{E}\|\nabla h(\theta_t)\|^2 - \frac{\eta}{\tau}\mathbb{E}\|\nabla h(\theta_t)\| + (\frac{\eta}{2\tau})^2 \leq \frac{1}{\sqrt{T}}\left[\frac{D}{\tau} + \frac{\beta\sigma^2}{2\tau}\right] + \frac{\eta\sigma}{\tau} + (\frac{\eta}{2\tau})^2 \\
\Leftrightarrow& \left|\mathbb{E}\|\nabla h(\theta_t)\| - \frac{\eta}{2\tau}\right| \leq \sqrt{\frac{1}{\sqrt{T}}\left[\frac{D}{\tau} + \frac{\beta\sigma^2}{2\tau}\right] + \frac{\eta\sigma}{\tau} + (\frac{\eta}{2\tau})^2}.
\end{aligned}$$

Then we remove the absolute value, and obtain

$$\mathbb{E}\|\nabla h(\theta_t)\| - \frac{\eta}{2\tau} \leq \sqrt{\frac{1}{\sqrt{T}}\left[\frac{D}{\tau} + \frac{\beta\sigma^2}{2\tau}\right] + \frac{\eta\sigma}{\tau} + (\frac{\eta}{2\tau})^2}$$

$$\Leftrightarrow \mathbb{E}\|\nabla h(\theta_t)\| \leq \frac{\eta}{2\tau} + \sqrt{\frac{1}{\sqrt{T}}\left[\frac{D}{\tau} + \frac{\beta\sigma^2}{2\tau}\right] + \frac{\eta\sigma}{\tau} + (\frac{\eta}{2\tau})^2}$$

$$\Leftrightarrow \mathbb{E}\|\nabla h(\theta_t)\|^2 \leq \left[\frac{\eta}{2\tau} + \sqrt{\frac{1}{\sqrt{T}}\left[\frac{D}{\tau} + \frac{\beta\sigma^2}{2\tau}\right] + \frac{\beta\sigma}{\tau} + (\frac{\eta}{2\tau})^2}\right]^2.$$

By Cauchy-Schwartz inequality, we have

$$\begin{aligned}
\mathbb{E}\|\nabla h(\theta_t)\|^2 \leq & \left[\frac{\eta}{2\tau} + \sqrt{\frac{1}{\sqrt{T}}\left[\frac{D}{\tau} + \frac{\beta\sigma^2}{2\tau}\right] + \frac{\eta\sigma}{\tau} + (\frac{\eta}{2\tau})^2}\right]^2 \\
\leq & 2\left[(\frac{\eta}{2\tau})^2 + \frac{1}{\sqrt{T}}\left[\frac{D}{\tau} + \frac{\beta\sigma^2}{2\tau}\right] + \frac{\beta\sigma}{\tau} + (\frac{\eta}{2\tau})^2\right] \\
= & \frac{\eta^2}{\tau^2} + \frac{2\eta\sigma}{\tau} + \frac{2}{\sqrt{T}}\left[\frac{D}{\tau} + \frac{\beta\sigma^2}{2\tau}\right] \\
= & \frac{\eta^2}{\tau^2} + \frac{2\eta\sigma}{\tau} + \mathcal{O}(\frac{1}{\sqrt{T}}).
\end{aligned}$$

□

In words, running SGD on an approximately smooth non-convex function, the algorithm cannot converge to a stationary point but with an additional constant term. Notice that this is an error bound for gradient norm. If we need an error bound for the optimality gap, we need an additional PL condition. Combining the optimization error and generalization error, we have

$$\mathcal{E}_{opt} + \mathcal{E}_{gen} \leq \mathcal{O}\left(\eta T + \frac{T}{n} + \frac{1}{\sqrt{T}}\right) + \text{constant},$$

where the first term is an additional term for adversarial training, which induces robust overfitting. Therefore, we can see that the analysis of the convex and non-convex cases do not have a major difference. To simplify the argument, we only discuss the convex case in the main paper.

B.2 Discussion on Strongly Convex Case

By (Nemirovski et al., 2009),

$$\mathcal{E}_{opt} \leq \frac{LD^2}{T} = \mathcal{O}(\frac{1}{T})$$

in the strongly convex case. Whether the function is smooth does not affect the convergence rate. Therefore,

$$\mathcal{E}_{opt} + \mathcal{E}_{gen} \leq \mathcal{O}\left(\eta + \frac{1}{n} + \frac{1}{T}\right).$$

This result shows that robust overfitting will disappear if the loss function is strongly convex. But the performance of adversarial training is still worse than the performance of standard training in $\mathcal{O}(\eta)$ in this strong assumption.

B.3 Discussion on Strongly Concave Assumption on the Inner Problem

In this subsection, we discuss the case that $g(\theta, z)$ is μ-strongly concave in z.

Assumption B.1. *The function g satisfies the following Lipschitzian smoothness conditions:*

$$\begin{aligned}
&\|g(\theta_1, z) - g(\theta_2, z)\| \leq L\|\theta_1 - \theta_2\|, \\
&\|\nabla_\theta g(\theta_1, z) - \nabla_\theta g(\theta_2, z)\| \leq L_\theta\|\theta_1 - \theta_2\|, \\
&\|\nabla_\theta g(\theta, z_1) - \nabla_\theta g(\theta, z_2)\| \leq L_z\|z_1 - z_2\|, \\
&\|\nabla_z g(\theta_1, z) - \nabla_\theta g(\theta_2, z)\| \leq L_{z\theta}\|\theta_1 - \theta_2\|.
\end{aligned}$$

Assumption B.1 assumes that the loss function is smooth (in zeroth-order and first-order), which are also used in the stability literature (Farnia and Ozdaglar, 2021; Xing et al., 2021a), as well as the convergence analysis literature (Wang et al., 2019; Liu et al., 2020). Comparing with Assumption 4.1, Assumption B.1 requires one more gradient Lipschitz $\|\nabla_z g(\theta_1, z) - \nabla_\theta g(\theta_2, z)\| \leq L_{z\theta}\|\theta_1 - \theta_2\|$.

Lemma B.1. *Under Assumption B.1, assume in addition that $g(\theta, z)$ is μ-strongly concave in z. $\forall \theta_1, \theta_2$ and $\forall z \in \mathcal{Z}$, the following properties hold.*

1. *(Lipschitz function.)* $\|h(\theta_1, z) - h(\theta_2, z)\| \leq L\|\theta_1 - \theta_2\|$.
2. *(gradient Lipschitz.)* $\|\nabla_\theta h(\theta_1, z) - \nabla_\theta h(\theta_2, z)\| \leq \beta_2\|\theta_1 - \theta_2\|$, *where*

$$\beta_2 = \frac{L_z L_{z\theta}}{\mu} + L_\theta.$$

The proof can be found in (Sinha et al., 2017; Wang et al., 2019). Therefore, the adversarial surrogate loss is β_2-gradient Lipschitz. The stability generalization bounds follows (Hardt et al., 2016) by replacing β by β_2 (for the choice of step size α).

C Additional Experiments

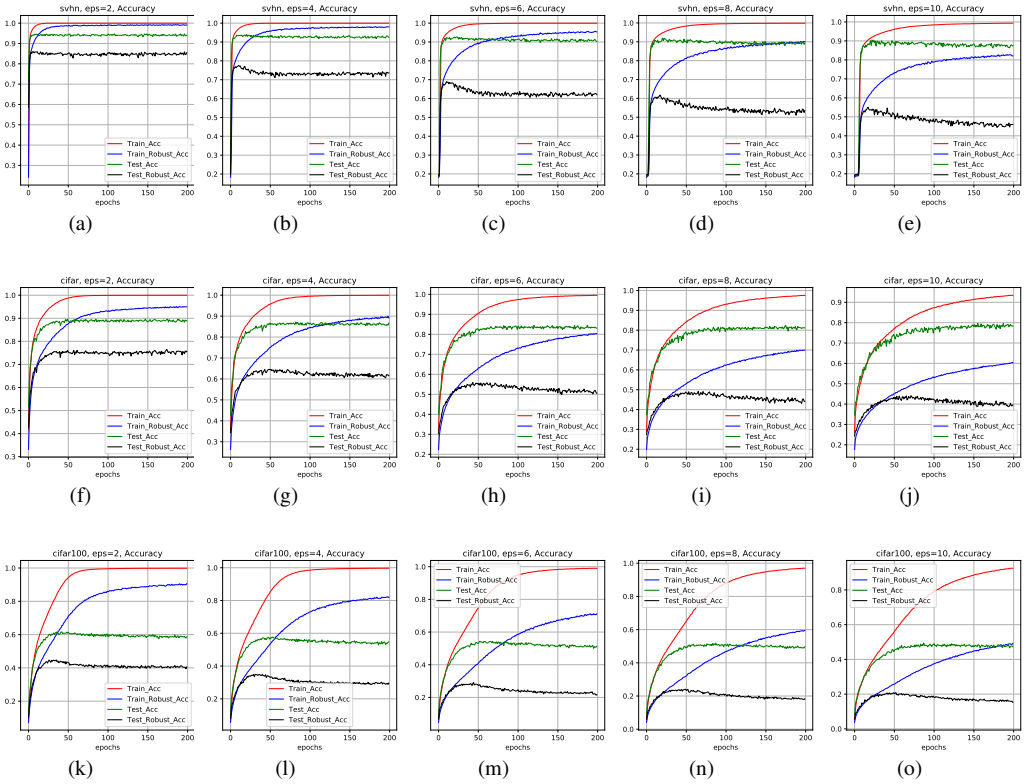


Figure 5: Accuracy of adversarial training with fixed learning rate = 0.01. The first row is the experiments on SVHN. The second row is the experiments on CIFAR-10. The last row is the experiments on CIFAR-100. The first column to the last column are the experiments of ϵ equal to 2, 4, 6, 8, and 10, respectively.

In this section, we provide additional experiments on SVHN, CIFAR-10, and CIFAR-100. In Fig. 5, we show the experiments of adversarial training using a fixed learning rate. In Fig. 6, we show the experiments of adversarial training using a standard piece-wise linear learning rate. In Fig. 7, we show the experiments of adversarial training using cyclic learning rate.

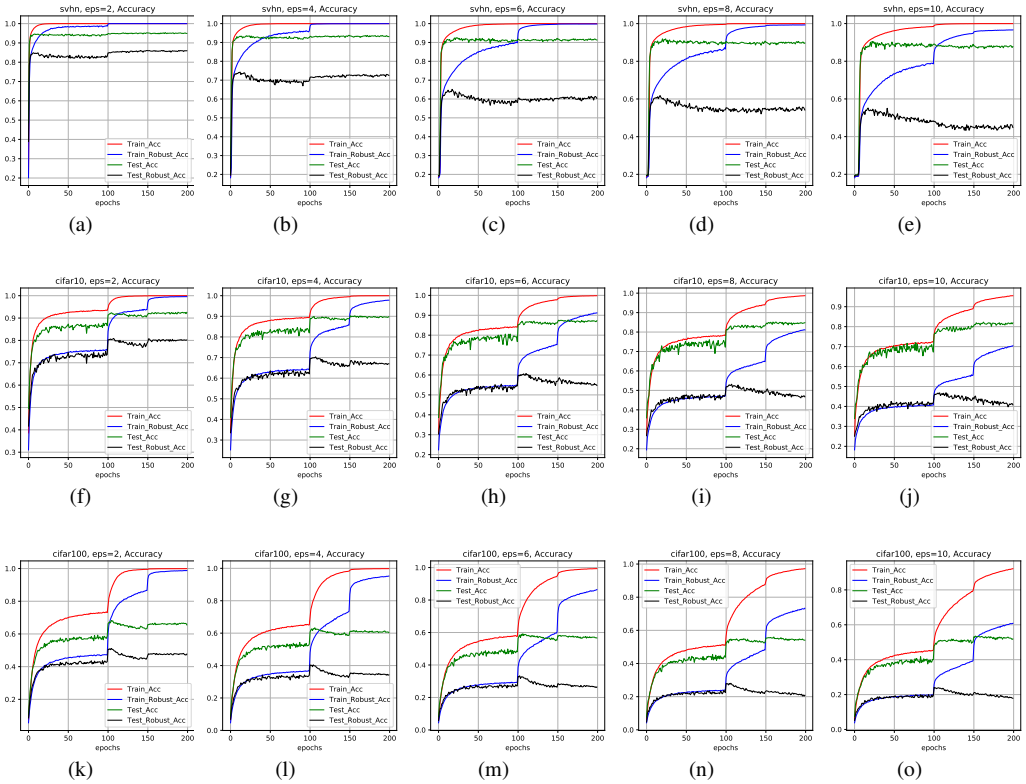


Figure 6: Accuracy of adversarial training with piece-wise linear learning rate. The first row is the experiments on SVHN. The second row is the experiments on CIFAR-10. The last row is the experiments on CIFAR-100. The first column to the last column are the experiments of ϵ equal to 2, 4, 6, 8, and 10, respectively.

Cyclic Learning Rate. We illustrate the experiments of adversarial training using cyclic learning rate. The learning rate increases linearly in the first 80 epochs and decreases to zero in the last 120 epochs. This learning rate mainly contributes to optimization. In terms of generalization, as discussed in theoretical settings, the generalization bound is no larger than that of the previous cases.

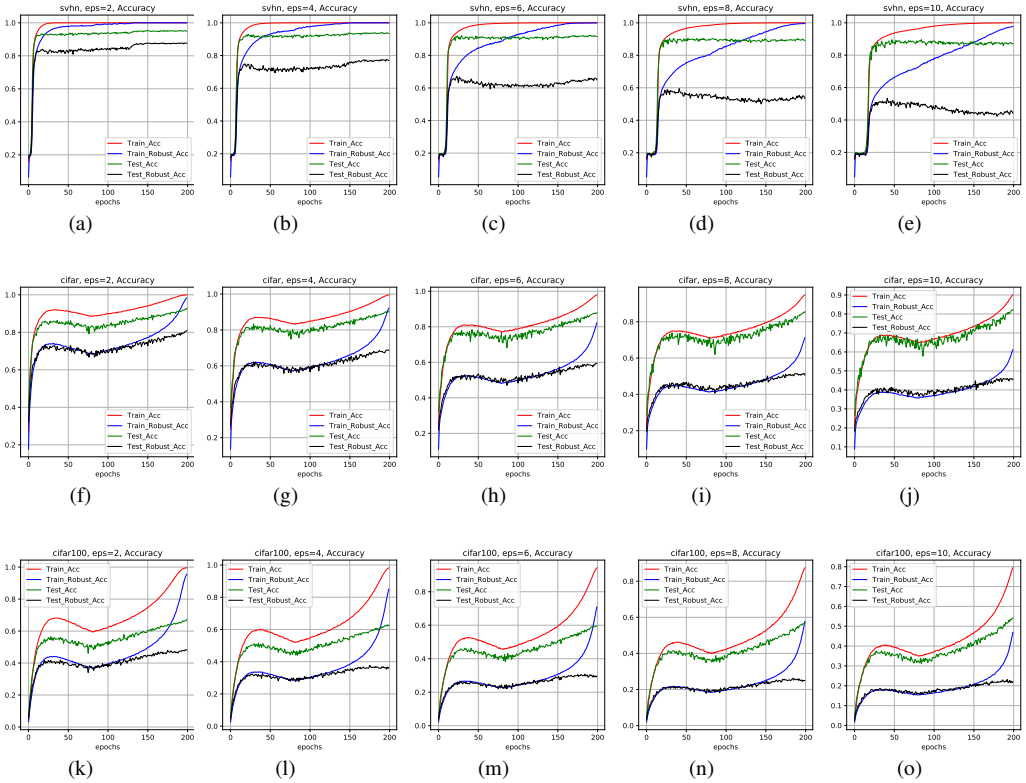


Figure 7: Accuracy of adversarial training with super-converge learning rate. The first row is the experiments on SVHN. The second row is the experiments on CIFAR-10. The last row is the experiments on CIFAR-100. The first column to the last column are the experiments of ϵ equal to 2, 4, 6, 8, and 10, respectively.