Rademacher Complexity for Adversarially Robust Generalization

Dong Yin [1] **Kannan Ramchandran** [1] **Peter Bartlett** [1 2]

Abstract

Many machine learning models are vulnerable to adversarial attacks; for example, adding adversarial perturbations that are imperceptible to humans can often make machine learning models produce wrong predictions with high confidence; moreover, although we may obtain robust models on the training dataset via adversarial training, in some problems the learned models cannot generalize well to the test data. In this paper, we focus on ℓ_∞ attacks, and study the adversarially robust generalization problem through the lens of Rademacher complexity. For binary linear classifiers, we prove tight bounds for the adversarial Rademacher complexity, and show that the adversarial Rademacher complexity is never smaller than its natural counterpart, and it has an unavoidable dimension dependence, unless the weight vector has bounded ℓ_1 norm, and our results also extend to multi-class linear classifiers; in addition, for (nonlinear) neural networks, we show that the dimension dependence in the adversarial Rademacher complexity also exists. We further consider a surrogate adversarial loss for one-hidden layer ReLU network and prove margin bounds for this setting. Our results indicate that having ℓ_1 norm constraints on the weight matrices might be a potential way to improve generalization in the adversarial setting. We demonstrate experimental results that validate our theoretical findings.

1. Introduction

In recent years, many modern machine learning models, in particular, deep neural networks, have achieved success in tasks such as image classification (He et al., 2016), speech recognition (Graves et al., 2013), machine translation (Bahdanau et al., 2014), game playing (Silver et al., 2016), etc. However, although these models achieve the state-of-the-art performance in many standard benchmarks or competitions, it has been observed that by adversarially adding some perturbation to the input of the model (images, audio signals), the machine learning models can make wrong predictions with high confidence. These adversarial inputs are often called the *adversarial examples*. Typical methods of generating adversarial examples include adding small perturbations that are imperceptible to humans (Szegedy et al., 2013), changing surrounding areas of the main objects in images (Gilmer et al., 2018a), and even simple rotation and translation (Engstrom et al., 2017). This phenomenon was first discovered by Szegedy et al. (2013) in image classification problems, and similar phenomena have been observed in other areas (Carlini & Wagner, 2018; Kos et al., 2018). Adversarial examples bring serious challenges in many security-critical applications, such as medical diagnosis and autonomous driving—the existence of these examples shows that many state-of-the-art machine learning models are actually unreliable in the presence of adversarial attacks.

Since the discovery of adversarial examples, there has been a race between designing robust models that can defend against adversarial attacks and designing attack algorithms that can generate adversarial examples and fool the machine learning models (Goodfellow et al., 2014; Gu & Rigazio, 2014; Carlini & Wagner, 2016; 2017). As of now, it seems that the attackers are winning this game. For example, a recent work shows that many of the defense algorithms fail when the attacker uses a carefully designed gradient-based method (Athalye et al., 2018). Meanwhile, *adversarial training* (Huang et al., 2015; Shaham et al., 2015; Madry et al., 2017) seems to be the most effective defense method. Adversarial training takes a robust optimization (Ben-Tal et al., 2009) perspective to the problem, and the basic idea is to minimize some *adversarial loss* over the training data. We elaborate below.

Suppose that data points $(\mathbf{x}, y)$ are drawn according to some unknown distribution $\mathcal{D}$ over the feature-label space $\mathcal{X} \times \mathcal{Y}$, and $\mathcal{X} \subseteq \mathbb{R}^d$. Let $\mathcal{F}$ be a hypothesis class (e.g., a class of neural networks with a particular architecture), and $\ell(f(\mathbf{x}), y)$ be the loss associated with $f \in \mathcal{F}$. Consider the ℓ_∞ white-box adversarial attack where an adversary is

[1] Department of Electrical Engineering and Computer Sciences, UC Berkeley, Berkeley, CA, USA [2] Department of Statistics, UC Berkeley, Berkeley, CA, USA. Correspondence to: Dong Yin <dongyin@eecs.berkeley.edu>.

Proceedings of the 36^{th} International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019. Copyright 2019 by the author(s).

allowed to observe the trained model and choose some $\mathbf{x}'$ such that $\|\mathbf{x}' - \mathbf{x}\|_\infty \le \epsilon$ and $\ell(f(\mathbf{x}'), y)$ is maximized. Therefore, to better defend against adversarial attacks, during training, the learner should aim to solve the empirical adversarial risk minimization problem

$$\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \max_{\|\mathbf{x}_i' - \mathbf{x}_i\|_\infty \le \epsilon} \ell(f(\mathbf{x}_i'), y_i), \qquad (1)$$

where $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ are n i.i.d. training examples drawn according to $\mathcal{D}$. This minimax formulation raises many interesting theoretical and practical questions. For example, we need to understand how to efficiently solve the optimization problem in (1), and in addition, we need to characterize the generalization property of the adversarial risk, i.e., the gap between the empirical adversarial risk in (1) and the population adversarial risk $\mathbb{E}_{(\mathbf{x},y)\sim D}[\max_{\|\mathbf{x}'-\mathbf{x}\|_\infty \le \epsilon} \ell(f(\mathbf{x}'), y)]$. In fact, for deep neural networks, both questions are still wide open. In particular, for the generalization problem, it has been observed that even if we can minimize the adversarial training error, the adversarial test error can still be large. For example, for a Resnet (He et al., 2016) model on CIFAR10, using the PGD adversarial training algorithm by Madry et al. (2017), one can achieve about 96% adversarial training accuracy, but the adversarial test accuracy is only 47%. This generalization gap is significantly larger than that in the natural setting (without adversarial attacks), and thus it has become increasingly important to better understand the generalization behavior of machine learning models in the adversarial setting.

In this paper, we focus on the adversarially robust generalization property and make a first step towards deeper understanding of this problem. We focus on ℓ_∞ adversarial attacks and analyze generalization through the lens of Rademacher complexity. We study both linear classifiers and nonlinear feedforward neural networks, and both binary and multi-class classification problems. We summarize our contributions as follows, and provide detailed comparison with existing works in Section 6.

1.1. Our Contributions

- For binary linear classifiers, we prove tight upper and lower bounds for the adversarial Rademacher complexity. We show that the adversarial Rademacher complexity is never smaller than its counterpart in the natural setting, which provides theoretical evidence for the empirical observation that adversarially robust generalization can be hard. We also show that under an ℓ_∞ adversarial attack, when the weight vector of the linear classifier has bounded ℓ_p norm $(p \ge 1)$, a polynomial dimension dependence in the adversarial Rademacher complexity is unavoidable, unless $p = 1$. For multi-class linear classifiers, we prove margin bounds in the adversarial setting. Similar to binary classifiers, the margin bound also exhibits polynomial dimension dependence when the weight vector for each class has bounded ℓ_p norm $(p > 1)$.
- For neural networks, we show that in contrast to the margin bounds derived by Bartlett et al. (2017) and Golowich et al. (2017) which depend only on the norms of the weight matrices and the data points, the adversarial Rademacher complexity has a lower bound with an explicit dimension dependence, which is also an effect of the ℓ_∞ attack. We further consider a *surrogate adversarial loss* for one hidden layer ReLU networks, based on the SDP relaxation proposed by Raghunathan et al. (2018a). We prove margin bounds using the surrogate loss and show that if the weight matrix of the first layer has bounded ℓ_1 norm, the margin bound does not have explicit dimension dependence. This suggests that in the adversarial setting, controlling the ℓ_1 norms of the weight matrices may be a way to improve generalization.
- We conduct experiments on linear classifiers and neural networks to validate our theoretical findings; more specifically, our experiments show that ℓ_1 regularization could reduce the adversarial generalization error, and the adversarial generalization gap increases as the dimension of the feature spaces increases.

Notation We define the set $[N] := \{1, 2, \ldots, N\}$. For two sets $\mathcal{A}$ and $\mathcal{B}$, we denote by $\mathcal{B}^{\mathcal{A}}$ the set of all functions from $\mathcal{A}$ to $\mathcal{B}$. We denote the indicator function of a event A as $\mathbb{1}(A)$. Unless otherwise stated, we denote vectors by boldface lowercase letters such as $\mathbf{w}$, and the elements in the vector are denoted by italics letters with subscripts, such as w_k. All-one vectors are denoted by $\mathbf{1}$. Matrices are denoted by boldface uppercase letters such as $\mathbf{W}$. For a matrix $\mathbf{W} \in \mathbb{R}^{d\times m}$ with columns $\mathbf{w}_i$, $i \in [m]$, the (p, q) matrix norm of $\mathbf{W}$ is defined as $\|\mathbf{W}\|_{p,q} = \|[\|\mathbf{w}_1\|_p, \|\mathbf{w}_2\|_p, \cdots, \|\mathbf{w}_m\|_p]\|_q$, and we may use the shorthand notation $\|\cdot\|_p \equiv \|\cdot\|_{p,p}$. We denote the spectral norm of matrices by $\|\cdot\|_\sigma$ and the Frobenius norm of matrices by $\|\cdot\|_F$ (i.e., $\|\cdot\|_F \equiv \|\cdot\|_2$). We use $\mathbb{B}_{\mathbf{x}}^\infty(\epsilon)$ to denote the ℓ_∞ ball centered at $\mathbf{x} \in \mathbb{R}^d$ with radius ϵ, i.e., $\mathbb{B}_{\mathbf{x}}^\infty(\epsilon) = \{\mathbf{x}' \in \mathbb{R}^d : \|\mathbf{x}' - \mathbf{x}\|_\infty \le \epsilon\}$.

2. Problem Setup

We start with the general statistical learning framework. Let $\mathcal{X}$ and $\mathcal{Y}$ be the feature and label spaces, respectively, and suppose that there is an unknown distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$. In this paper, we assume that the feature space is a subset of the d dimensional Euclidean space, i.e., $\mathcal{X} \subseteq \mathbb{R}^d$. Let $\mathcal{F} \subseteq \mathcal{V}^{\mathcal{X}}$ be the hypothesis class that we use to make predictions, where $\mathcal{V}$ is another space that might be different from $\mathcal{Y}$. Let $\ell : \mathcal{V} \times \mathcal{Y} \to [0, B]$ be the loss function. Throughout this paper we assume that ℓ is bounded, i.e., B is a positive constant. In addition, we introduce the function class $\ell_{\mathcal{F}} \subseteq [0, B]^{\mathcal{X}\times\mathcal{Y}}$ by composing the functions in $\mathcal{F}$ with $\ell(\cdot, y)$, i.e., $\ell_{\mathcal{F}} := \{(\mathbf{x}, y) \mapsto \ell(f(\mathbf{x}), y) : f \in \mathcal{F}\}$. The goal of the learning problem is to find $f \in \mathcal{F}$ such that the *population risk* $R(f) := \mathbb{E}_{(\mathbf{x},y)\in\mathcal{D}}[\ell(f(\mathbf{x}),y)]$ is minimized.

We consider the supervised learning setting where one has access to n i.i.d. training examples drawn according to $\mathcal{D}$, denoted by $(\mathbf{x}_1,y_1),(\mathbf{x}_2,y_2),\dots,(\mathbf{x}_n,y_n)$. A learning algorithm maps the n training examples to a hypothesis $f\in\mathcal{F}$. In this paper, we are interested in the gap between the *empirical risk* $R_n(f) := \frac{1}{n}\sum_{i=1}^n \ell(f(\mathbf{x}_i),y_i)$ and the population risk $R(f)$, known as the generalization error.

Rademacher complexity (Bartlett & Mendelson, 2002) is one of the classic measures of generalization error. Here, we present its formal definition. For any function class $\mathcal{H}\subseteq\mathbb{R}^{\mathcal{Z}}$, given a sample $\mathcal{S}=\{\mathbf{z}_1,\mathbf{z}_2,\dots,\mathbf{z}_n\}$ of size n, and *empirical Rademacher complexity* is defined as

$$\mathfrak{R}_{\mathcal{S}}(\mathcal{H}) := \frac{1}{n}\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{h\in\mathcal{H}}\sum_{i=1}^n \sigma_i h(\mathbf{z}_i)\right],$$

where $\sigma_1,\dots,\sigma_n$ are i.i.d. Rademacher random variables with $\mathbb{P}\{\sigma_i=1\}=\mathbb{P}\{\sigma_i=-1\}=\frac{1}{2}$. In our learning problem, denote the training sample by $\mathcal{S}$, i.e., $\mathcal{S}:=\{(\mathbf{x}_1,y_1),(\mathbf{x}_2,y_2),\dots,(\mathbf{x}_n,y_n)\}$. We then have the following theorem which connects the population and empirical risks via Rademacher complexity.

Theorem 1. *(Bartlett & Mendelson, 2002; Mohri et al., 2012) Suppose that the range of $\ell(f(\mathbf{x}),y)$ is $[0,B]$. Then, for any $\delta\in(0,1)$, with probability at least $1-\delta$, the following holds for all $f\in\mathcal{F}$,*

$$R(f)\le R_n(f)+2B\mathfrak{R}_{\mathcal{S}}(\ell_{\mathcal{F}})+3B\sqrt{\frac{\log\frac{2}{\delta}}{2n}}.$$

As we can see, Rademacher complexity measures the rate that the empirical risk converges to the population risk *uniformly* across $\mathcal{F}$. In fact, according to the anti-symmetrization lower bound by Koltchinskii et al. (2006), one can show that $\mathfrak{R}_{\mathcal{S}}(\ell_{\mathcal{F}})\lesssim\sup_{f\in\mathcal{F}}R(f)-R_n(f)\lesssim\mathfrak{R}_{\mathcal{S}}(\ell_{\mathcal{F}})$. Therefore, Rademacher complexity is a tight bound for the rate of uniform convergence of a loss function class, and in many settings can be a tight bound for generalization error.

The above discussions can be extended to the adversarial setting. In this paper, we focus on the ℓ_∞ adversarial attack. In this setting, the learning algorithm still has access to n i.i.d. uncorrupted training examples drawn according to $\mathcal{D}$. Once the learning procedure finishes, the output hypothesis f is revealed to an adversary. For any data point $(\mathbf{x},y)$ drawn according to $\mathcal{D}$, the adversary is allowed to perturb $\mathbf{x}$ within some ℓ_∞ ball to maximize the loss. Our goal is to minimize the *adversarial population risk*, i.e.,

$$\widetilde{R}(f) := \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\max_{\mathbf{x}'\in\mathbb{B}_{\mathbf{x}}^\infty(\epsilon)}\ell(f(\mathbf{x}'),y)\right],$$

and to this end, a natural way is to conduct *adversarial training*—minimizing the adversarial empirical risk

$$\widetilde{R}_n(f) := \frac{1}{n}\sum_{i=1}^n \max_{\mathbf{x}_i'\in\mathbb{B}_{\mathbf{x}_i}^\infty(\epsilon)}\ell(f(\mathbf{x}_i'),y_i).$$

Let us define the adversarial loss $\widetilde{\ell}(f(\mathbf{x}),y) := \max_{\mathbb{B}_{\mathbf{x}}^\infty(\epsilon)}\ell(f(\mathbf{x}'),y)$ and the function class $\widetilde{\ell}_{\mathcal{F}}\subseteq[0,B]^{\mathcal{X}\times\mathcal{Y}}$ as $\widetilde{\ell}_{\mathcal{F}} := \{\widetilde{\ell}(f(\mathbf{x}),y): f\in\mathcal{F}\}$. Since the range of $\widetilde{\ell}(f(\mathbf{x}),y)$ is still $[0,B]$, we have the following direct corollary of Theorem 1.

Corollary 1. *For any $\delta\in(0,1)$, with probability at least $1-\delta$, the following holds for all $f\in\mathcal{F}$,*

$$\widetilde{R}(f)\le\widetilde{R}_n(f)+2B\mathfrak{R}_{\mathcal{S}}(\widetilde{\ell}_{\mathcal{F}})+3B\sqrt{\frac{\log\frac{2}{\delta}}{2n}}.$$

As we can see, the Rademacher complexity of the adversarial loss function class $\widetilde{\ell}_{\mathcal{F}}$, i.e., $\mathfrak{R}_{\mathcal{S}}(\widetilde{\ell}_{\mathcal{F}})$ is again the key quantity for the generalization ability of the learning problem. A natural problem of interest is to compare the Rademacher complexities in the natural and the adversarial settings, and obtain generalization bounds for the adversarial loss.

3. Linear Classifiers

3.1. Binary Classification

We start with binary linear classifiers. In this setting, we define $\mathcal{Y}=\{-1,+1\}$, and let the hypothesis class $\mathcal{F}\subseteq\mathbb{R}^{\mathcal{X}}$ be a set of linear functions of $\mathbf{x}\in\mathcal{X}$. More specifically, we define $f_{\mathbf{w}}(\mathbf{x}) := \langle\mathbf{w},\mathbf{x}\rangle$, and consider prediction vector $\mathbf{w}$ with ℓ_p norm constraint ($p\ge 1$), i.e.,

$$\mathcal{F}=\{f_{\mathbf{w}}(\mathbf{x}):\|\mathbf{w}\|_p\le W\}. \tag{2}$$

We predict the label with the sign of $f_{\mathbf{w}}(\mathbf{x})$; more specifically, we assume that the loss function $\ell(f_{\mathbf{w}}(\mathbf{x}),y)$ can be written as $\ell(f_{\mathbf{w}}(\mathbf{x}),y)\equiv\phi(y\langle\mathbf{w},\mathbf{x}\rangle)$, where $\phi:\mathbb{R}\to[0,B]$ is monotonically nonincreasing and L_ϕ-Lipschitz. In fact, if $\phi(0)\ge 1$, we can obtain a bound on the classification error according to Theorem 1. That is, with probability at least $1-\delta$, for all $f_{\mathbf{w}}\in\mathcal{F}$,

$$\begin{aligned}&\mathbb{P}_{(\mathbf{x},y)\sim\mathcal{D}}\{\operatorname{sgn}(f_{\mathbf{w}}(\mathbf{x}))\neq y\}\\ &\le\frac{1}{n}\sum_{i=1}^n\ell(f_{\mathbf{w}}(\mathbf{x}_i),y_i)+2B\mathfrak{R}_{\mathcal{S}}(\ell_{\mathcal{F}})+3B\sqrt{\frac{\log\frac{2}{\delta}}{2n}}.\end{aligned}$$

In addition, recall that according to the Ledoux-Talagrand contraction inequality (Ledoux & Talagrand, 2013), we have $\mathfrak{R}_{\mathcal{S}}(\ell_{\mathcal{F}})\le L_\phi\mathfrak{R}_{\mathcal{S}}(\mathcal{F})$.

For the adversarial setting, we have

$$\widetilde{\ell}(f_{\mathbf{w}}(\mathbf{x}),y)=\max_{\mathbf{x}'\in\mathbb{B}_{\mathbf{x}}^\infty(\epsilon)}\ell(f_{\mathbf{w}}(\mathbf{x}'),y)=\phi(\min_{\mathbf{x}'\in\mathbb{B}_{\mathbf{x}}^\infty(\epsilon)}y\langle\mathbf{w},\mathbf{x}'\rangle).$$

Let us define the following function class $\widetilde{\mathcal{F}}\subseteq\mathbb{R}^{\mathcal{X}\times\{\pm1\}}$:

$$\widetilde{\mathcal{F}}=\left\{\min_{\mathbf{x}'\in\mathbb{B}_{\mathbf{x}}^\infty(\epsilon)}y\langle\mathbf{w},\mathbf{x}'\rangle:\|\mathbf{w}\|_p\le W\right\}. \tag{3}$$

Again, we have $\mathfrak{R}_{\mathcal{S}}(\widetilde{\ell}_{\mathcal{F}})\le L_\phi\mathfrak{R}_{\mathcal{S}}(\widetilde{\mathcal{F}})$. The first major contribution of our work is the following theorem, which provides a comparison between $\mathfrak{R}_{\mathcal{S}}(\mathcal{F})$ and $\mathfrak{R}_{\mathcal{S}}(\widetilde{\mathcal{F}})$.

Theorem 2 **(Main Result 1).** *Let $\mathcal{F} := \{f_{\mathbf{w}}(\mathbf{x}) : \|\mathbf{w}\|_p \le W\}$ and $\widetilde{\mathcal{F}} := \{\min_{\mathbf{x}'\in\mathbb{B}_{\mathbf{x}}^{\infty}(\epsilon)} y\langle \mathbf{w}, \mathbf{x}'\rangle : \|\mathbf{w}\|_p \le W\}$. Suppose that $\frac{1}{p} + \frac{1}{q} = 1$. Then, there exists a universal constant $c \in (0,1)$ such that*

$$\max\{\mathfrak{R}_{\mathcal{S}}(\mathcal{F}), c\epsilon W \frac{d^{\frac{1}{q}}}{\sqrt{n}}\} \le \mathfrak{R}_{\mathcal{S}}(\widetilde{\mathcal{F}}) \le \mathfrak{R}_{\mathcal{S}}(\mathcal{F}) + \epsilon W \frac{d^{\frac{1}{q}}}{\sqrt{n}}.$$

We prove Theorem 2 in Appendix B. We can see that the adversarial Rademacher complexity, i.e., $\mathfrak{R}_{\mathcal{S}}(\widetilde{\mathcal{F}})$ is always at least as large as the Rademacher complexity in the natural setting. This implies that uniform convergence in the adversarial setting is at least as hard as that in the natural setting. In addition, since $\max\{a, b\} \ge \frac{1}{2}(a + b)$, we have

$$\frac{c}{2}\Big(\mathfrak{R}_{\mathcal{S}}(\mathcal{F}) + \epsilon W \frac{d^{\frac{1}{q}}}{\sqrt{n}}\Big) \le \mathfrak{R}_{\mathcal{S}}(\widetilde{\mathcal{F}}) \le \mathfrak{R}_{\mathcal{S}}(\mathcal{F}) + \epsilon W \frac{d^{\frac{1}{q}}}{\sqrt{n}}.$$

Therefore, we have a tight bound for $\mathfrak{R}_{\mathcal{S}}(\widetilde{\mathcal{F}})$ up to a constant factor. Further, if $p > 1$ the adversarial Rademacher complexity has an unavoidable polynomial dimension dependence, i.e., $\mathfrak{R}_{\mathcal{S}}(\widetilde{\mathcal{F}})$ is always at least as large as $\mathcal{O}(\epsilon W \frac{d^{1/q}}{\sqrt{n}})$. On the other hand, one can easily show that in the natural setting, $\mathfrak{R}_{\mathcal{S}}(\mathcal{F}) = \frac{W}{n}\mathbb{E}_{\boldsymbol{\sigma}}[\|\sum_{i=1}^{n} \sigma_i \mathbf{x}_i\|_q]$, which implies that $\mathfrak{R}_{\mathcal{S}}(\mathcal{F})$ depends on the distribution of $\mathbf{x}_i$ and the norm constraint W, but does not have an explicit dimension dependence. This means that $\mathfrak{R}_{\mathcal{S}}(\widetilde{\mathcal{F}})$ could be order-wise larger than $\mathfrak{R}_{\mathcal{S}}(\mathcal{F})$, depending on the distribution of the data. An interesting fact is that, if we have an ℓ_1 norm constraint on the prediction vector $\mathbf{w}$, we can avoid the dimension dependence in $\mathfrak{R}_{\mathcal{S}}(\widetilde{\mathcal{F}})$.

3.2. Multi-class Classification

Margin Bounds for Multi-class Classification We proceed to study multi-class linear classifiers. We start with the standard margin bound framework for multi-class classification. In K-class classification problems, we choose $\mathcal{Y} = [K]$, and the functions in the hypothesis class $\mathcal{F}$ map $\mathcal{X}$ to $\mathbb{R}^K$, i.e., $\mathcal{F} \subseteq (\mathbb{R}^K)^{\mathcal{X}}$. Intuitively, the k-th coordinate of $f(\mathbf{x})$ is the score that f gives to the k-th class, and we make prediction by choosing the class with the highest score. We define the margin operator $M(\mathbf{z}, y) : \mathbb{R}^K \times [K] \to \mathbb{R}$ as $M(\mathbf{z}, y) = z_y - \max_{y'\ne y} z_{y'}$. For a training example $(\mathbf{x}, y)$, a hypothesis f makes a correct prediction if and only if $M(f(\mathbf{x}), y) > 0$. We also define function class $M_{\mathcal{F}} := \{(\mathbf{x}, y) \mapsto M(f(\mathbf{x}), y) : f \in \mathcal{F}\} \subseteq \mathbb{R}^{\mathcal{X}\times[K]}$. For multi-class classification problems, we consider a particular loss function $\ell(f(\mathbf{x}), y) = \phi_\gamma(M(f(\mathbf{x}), y))$, where $\gamma > 0$ and $\phi_\gamma : \mathbb{R} \to [0, 1]$ is the ramp loss:

$$\phi_\gamma(t) = \begin{cases} 1 & t \le 0 \\ 1 - \frac{t}{\gamma} & 0 < t < \gamma \\ 0 & t \ge \gamma. \end{cases} \tag{4}$$

One can check that $\ell(f(\mathbf{x}), y)$ satisfies:

$$\begin{aligned} &\mathbb{1}(y \ne \arg\max_{y'\in[K]} [f(\mathbf{x})]_{y'}) \le \ell(f(\mathbf{x}), y) \\ &\le \mathbb{1}([f(\mathbf{x})]_y \le \gamma + \max_{y'\ne y}[f(\mathbf{x})]_{y'}). \end{aligned} \tag{5}$$

Let $\mathcal{S} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n} \in (\mathcal{X} \times [K])^n$ be the i.i.d. training examples, and define the function class $\ell_{\mathcal{F}} := \{(\mathbf{x}, y) \mapsto \phi_\gamma(M(f(\mathbf{x}), y)) : f \in \mathcal{F}\} \subseteq \mathbb{R}^{\mathcal{X}\times[K]}$. Since $\phi_\gamma(t) \in [0, 1]$ and $\phi_\gamma(\cdot)$ is $1/\gamma$-Lipschitz, by combining (5) with Theorem 1, we can obtain the following direct corollary as the generalization bound in the multi-class setting (Mohri et al., 2012).

Corollary 2. *Consider the above multi-class classification setting. For any fixed $\gamma > 0$, we have with probability at least $1 - \delta$, for all $f \in \mathcal{F}$,*

$$\begin{aligned} &\mathbb{P}_{(\mathbf{x},y)\sim\mathcal{D}}\Big\{y \ne \arg\max_{y'\in[K]} [f(\mathbf{x})]_{y'}\Big\} \\ \le &\frac{1}{n}\sum_{i=1}^{n} \mathbb{1}([f(\mathbf{x}_i)]_{y_i} \le \gamma + \max_{y'\ne y}[f(\mathbf{x}_i)]_{y'}) + 2\mathfrak{R}_{\mathcal{S}}(\ell_{\mathcal{F}}) \\ &+ 3\sqrt{\frac{\log\frac{2}{\delta}}{2n}}. \end{aligned}$$

In the adversarial setting, the adversary tries to maximize the loss $\ell(f(\mathbf{x}), y) = \phi_\gamma(M(f(\mathbf{x}), y))$ around an ℓ_∞ ball centered at $\mathbf{x}$. We have the adversarial loss $\widetilde{\ell}(f(\mathbf{x}), y) := \max_{\mathbf{x}'\in\mathbb{B}_{\mathbf{x}}^{\infty}(\epsilon)} \ell(f(\mathbf{x}'), y)$, and the function class $\widetilde{\ell}_{\mathcal{F}} := \{(\mathbf{x}, y) \mapsto \widetilde{\ell}(f(\mathbf{x}), y) : f \in \mathcal{F}\} \subseteq \mathbb{R}^{\mathcal{X}\times[K]}$. Thus, we have the following generalization bound in the adversarial setting.

Corollary 3. *Consider the above adversarial multi-class classification setting. For any fixed $\gamma > 0$, we have with probability at least $1 - \delta$, for all $f \in \mathcal{F}$,*

$$\begin{aligned} &\mathbb{P}_{(\mathbf{x},y)\sim\mathcal{D}}\Big\{\exists\, \mathbf{x}' \in \mathbb{B}_{\mathbf{x}}^{\infty}(\epsilon) \textit{ s.t. } y \ne \arg\max_{y'\in[K]} [f(\mathbf{x}')]_{y'}\Big\} \\ \le &\frac{1}{n}\sum_{i=1}^{n} \mathbb{1}(\exists\, \mathbf{x}_i' \in \mathbb{B}_{\mathbf{x}_i}^{\infty}(\epsilon), [f(\mathbf{x}_i')]_{y_i} \le \gamma + \max_{y'\ne y}[f(\mathbf{x}_i')]_{y'}) \\ &+ 2\mathfrak{R}_{\mathcal{S}}(\widetilde{\ell}_{\mathcal{F}}) + 3\sqrt{\frac{\log\frac{2}{\delta}}{2n}}. \end{aligned}$$

Multi-class Linear Classifiers We now focus on multi-class linear classifiers. For linear classifiers, each function in the hypothesis class is linearly parametrized by a matrix $\mathbf{W} \in \mathbb{R}^{K\times d}$, i.e., $f(\mathbf{x}) \equiv f_{\mathbf{W}}(\mathbf{x}) = \mathbf{W}\mathbf{x}$. Let $\mathbf{w}_k \in \mathbb{R}^d$ be the k-th column of $\mathbf{W}^\top$, then we have $[f_{\mathbf{W}}(\mathbf{x})]_k = \langle \mathbf{w}_k, \mathbf{x}\rangle$. We assume that each $\mathbf{w}_k$ has ℓ_p norm $(p \ge 1)$ upper bounded by W, which implies that $\mathcal{F} = \{f_{\mathbf{W}}(\mathbf{x}) : \|\mathbf{W}^\top\|_{p,\infty} \le W\}$. In the natural setting, we have the following margin bound for linear classifiers as a corollary of the multi-class margin bounds by Kuznetsov et al. (2015); Maximov & Reshetova (2016).

Theorem 3. *Consider the multi-class linear classifiers in the above setting, and suppose that $\frac{1}{p} + \frac{1}{q} = 1$, $p, q \ge 1$. For any fixed $\gamma > 0$ and $W > 0$, we have with probability at least $1 - \delta$, for all $\mathbf{W}$ such that $\|\mathbf{W}^\top\|_{p,\infty} \le W$,*

$$\begin{aligned}&\mathbb{P}_{(\mathbf{x},y)\sim\mathcal{D}}\left\{y\neq\arg\max_{y'\in[K]}\langle\mathbf{w}_{y'},\mathbf{x}\rangle\right\}\\ \leq&\frac{1}{n}\sum_{i=1}^{n}\mathbb{1}(\langle\mathbf{w}_{y_i},\mathbf{x}_i\rangle\leq\gamma+\max_{y'\neq y_i}\langle\mathbf{w}_{y'},\mathbf{x}_i\rangle)\\&+\frac{4KW}{\gamma n}\mathbb{E}_{\boldsymbol{\sigma}}\left[\|\sum_{i=1}^{n}\sigma_i\mathbf{x}_i\|_q\right]+3\sqrt{\frac{\log\frac{2}{\delta}}{2n}}.\end{aligned}$$

We prove Theorem 3 in Appendix C.1 for completeness. In the adversarial setting, we have the following margin bound.

Theorem 4 (**Main Result 2**). *Consider the multi-class linear classifiers in the adversarial setting, and suppose that $\frac{1}{p}+\frac{1}{q}=1$, $p,q\geq 1$. For any fixed $\gamma>0$ and $W>0$, we have with probability at least $1-\delta$, for all $\mathbf{W}$ such that $\|\mathbf{W}^\top\|_{p,\infty}\leq W$,*

$$\begin{aligned}&\mathbb{P}_{(\mathbf{x},y)\sim\mathcal{D}}\Big\{\exists\,\mathbf{x}'\in\mathbb{B}_{\mathbf{x}}^{\infty}(\epsilon),\ s.t.\ y\neq\arg\max_{y'\in[K]}\langle\mathbf{w}_{y'},\mathbf{x}\rangle\Big\}\\ \leq&\frac{1}{n}\sum_{i=1}^{n}E_i+\frac{2WK}{\gamma}\Big[\frac{\epsilon\sqrt{K}d^{\frac{1}{q}}}{\sqrt{n}}+\frac{1}{n}\sum_{y=1}^{K}U_y\Big]+3\sqrt{\frac{\log\frac{2}{\delta}}{2n}},\end{aligned}$$

where

$$E_i=\mathbb{1}(\langle\mathbf{w}_{y_i},\mathbf{x}_i\rangle\leq\gamma+\max_{y'\neq y_i}(\langle\mathbf{w}_{y'},\mathbf{x}_i\rangle+\epsilon\|\mathbf{w}_{y'}-\mathbf{w}_{y_i}\|_1)),$$

$$U_y=\mathbb{E}_{\boldsymbol{\sigma}}\Big[\|\sum_{i=1}^{n}\sigma_i\mathbf{x}_i\mathbb{1}(y_i=y)\|_q\Big].$$

We prove Theorem 4 in Appendix C.2. As we can see, similar to the binary classification problems, if $p>1$, the margin bound in the adversarial setting has an explicit polynomial dependence on d, whereas in the natural setting, the margin bound does not have dimension dependence. This shows that, at least for the generalization upper bound that we obtain, the dimension dependence in the adversarial setting also exists in the multi-class classification problems.

4. Neural Networks

We proceed to consider feedforward neural networks with ReLU activation. Here, each function f in the hypothesis class $\mathcal{F}$ is parametrized by a sequence of matrices $\mathbf{W}=(\mathbf{W}_1,\mathbf{W}_2,\ldots,\mathbf{W}_L)$, i.e., $f\equiv f_{\mathbf{W}}$. Assume that $\mathbf{W}_h\in\mathbb{R}^{d_h\times d_{h-1}}$, and $\rho(\cdot)$ be the ReLU function, i.e., $\rho(t)=\max\{t,0\}$ for $t\in\mathbb{R}$. For vectors, $\rho(\mathbf{x})$ is vector generated by applying $\rho(\cdot)$ on each coordinate of $\mathbf{x}$, i.e., $[\rho(\mathbf{x})]_i=\rho(x_i)$. We have[1]

$$f_{\mathbf{W}}(\mathbf{x})=\mathbf{W}_L\rho(\mathbf{W}_{L-1}\rho(\cdots\rho(\mathbf{W}_1\mathbf{x})\cdots)).$$

For K-class classification, we have $d_L=K$, $f_{\mathbf{W}}(\mathbf{x}):\mathbb{R}^d\to\mathbb{R}^K$, and $[f_{\mathbf{W}}(\mathbf{x})]_k$ is the score for the k-th class. In the special case of binary classification, as discussed in Section 3.1, we can have $\mathcal{Y}=\{+1,-1\}$, $d_L=1$, and the loss function can be written as

$$\ell(f_{\mathbf{W}}(\mathbf{x}),y)=\phi(yf_{\mathbf{W}}(\mathbf{x})),$$

where $\phi:\mathbb{R}\to[0,B]$ is monotonically nonincreasing and L_ϕ-Lipschitz.

4.1. Comparison of Rademacher Complexity Bounds

We start with a comparison of Rademacher complexities of neural networks in the natural and adversarial settings. Although naively applying the definition of Rademacher complexity may provide a loose generalization bound (Zhang et al., 2016a), when properly normalized by the margin, one can still derive generalization bound that matches experimental observations via Rademacher complexity (Bartlett et al., 2017). Our comparison shows that, when the weight matrices of the neural networks have bounded norms, in the natural setting, the Rademacher complexity is upper bounded by a quantity which only has logarithmic dependence on the dimension; however, in the adversarial setting, the Rademacher complexity is lower bounded by a quantity with explicit $\sqrt{d}$ dependence.

We focus on the binary classification. For the natural setting, we review the results by Bartlett et al. (2017). Let $\mathcal{S}=\{(\mathbf{x}_i,y_i)\}_{i=1}^n\in(\mathcal{X}\times\{-1,+1\})^n$ be the i.i.d. training examples, and define $\mathbf{X}:=[\mathbf{x}_1,\mathbf{x}_2,\cdots,\mathbf{x}_n]\in\mathbb{R}^{d\times n}$, and $d_{\max}=\max\{d,d_1,d_2,\ldots,d_L\}$.

Theorem 5. *(Bartlett et al., 2017) Consider the neural network hypothesis class $\mathcal{F}=\{f_{\mathbf{W}}(\mathbf{x}):\mathbf{W}=(\mathbf{W}_1,\mathbf{W}_2,\ldots,\mathbf{W}_L),\|\mathbf{W}_h\|_\sigma\leq s_h,\|\mathbf{W}_h^\top\|_{2,1}\leq b_h,h\in[L]\}\subseteq\mathbb{R}^{\mathcal{X}}$. Then, we have*

$$\mathfrak{R}_{\mathcal{S}}(\mathcal{F})\leq\frac{4}{n^{3/2}}+\frac{26\log(n)\log(2d_{\max})}{n}A,$$

where $A=\|\mathbf{X}\|_F\Big(\prod_{h=1}^{L}s_h\Big)\Big(\sum_{j=1}^{L}(\frac{b_j}{s_j})^{2/3}\Big)^{3/2}$.

On the other hand, in this work, we prove the following result which shows that when the product of the spectral norms of all the weight matrices is bounded, the Rademacher complexity of the adversarial loss function class is lower bounded by a quantity with an explicit $\sqrt{d}$ factor. More specifically, for binary classification problems, since

$$\widetilde{\ell}(f_{\mathbf{W}}(\mathbf{x}),y)=\max_{\mathbf{x}'\in\mathbb{B}_{\mathbf{x}}^{\infty}(\epsilon)}\ell(f_{\mathbf{W}}(\mathbf{x}'),y)=\phi(\min_{\mathbf{x}'\in\mathbb{B}_{\mathbf{x}}^{\infty}(\epsilon)}yf_{\mathbf{W}}(\mathbf{x}')),$$

and $\phi(\cdot)$ is Lipschitz, we consider the function class

$$\begin{aligned}\widetilde{\mathcal{F}}=\{(\mathbf{x},y)\mapsto&\min_{\mathbf{x}'\in\mathbb{B}_{\mathbf{x}}^{\infty}(\epsilon)}yf_{\mathbf{W}}(\mathbf{x}'):\\ &\mathbf{W}=(\mathbf{W}_1,\mathbf{W}_2,\ldots,\mathbf{W}_L),\prod_{h=1}^{L}\|\mathbf{W}_h\|_\sigma\leq r\}\subseteq\mathbb{R}^{\mathcal{X}\times\{\pm1\}}.\end{aligned}\tag{6}$$

Then we have the following result.

Theorem 6 (**Main Result 3**). *Let $\widetilde{\mathcal{F}}$ be defined as in (6). Then, there exists a universal constant $c>0$ such that*

$$\mathfrak{R}_{\mathcal{S}}(\widetilde{\mathcal{F}})\geq cr\Big(\frac{1}{n}\|\mathbf{X}\|_F+\epsilon\sqrt{\frac{d}{n}}\Big).$$

[1] This implies that $d_0\equiv d$.

We prove Theorem 6 in Appendix D.1. This result shows that if we aim to study the Rademacher complexity of the function class defined as in (6), a $\sqrt{d}$ dimension dependence may be unavoidable, in contrast to the natural setting where the dimension dependence is only logarithmic.

4.2. Generalization Bound for Surrogate Adversarial Loss

For neural networks, even if there is only one hidden layer, for a particular data point $(\mathbf{x}, y)$, evaluating the adversarial loss $\widetilde{\ell}(f_{\mathbf{W}}(\mathbf{x}), y) = \max_{\mathbf{x}' \in \mathbb{B}_{\mathbf{x}}^{\infty}(\epsilon)} \ell(f_{\mathbf{W}}(\mathbf{x}'), y)$ can be hard, since it requires maximizing a non-concave function in a bounded set. A recent line of work tries to find upper bounds for $\widetilde{\ell}(f_{\mathbf{W}}(\mathbf{x}), y)$ that can be computed in polynomial time. More specifically, we would like to find *surrogate adversarial loss* $\widehat{\ell}(f_{\mathbf{W}}(\mathbf{x}), y)$ such that $\widehat{\ell}(f_{\mathbf{W}}(\mathbf{x}), y) \geq \widetilde{\ell}(f_{\mathbf{W}}(\mathbf{x}), y)$, $\forall\ \mathbf{x}, y, \mathbf{W}$. These surrogate adversarial loss can thus provide *certified* defense against adversarial examples, and can be computed efficiently. In addition, the surrogate adversarial loss $\widehat{\ell}(f_{\mathbf{W}}(\mathbf{x}), y)$ should be as tight as possible—it should be close enough to the original adversarial loss $\widetilde{\ell}(f_{\mathbf{W}}(\mathbf{x}), y)$, so that the surrogate adversarial loss can indeed represent the robustness of the model against adversarial attacks. Recently, a few approaches to designing surrogate adversarial loss have been developed, and SDP relaxation (Raghunathan et al., 2018a;b) and LP relaxation (Kolter & Wong, 2017; Wong et al., 2018) are two major examples.

In this section, we focus on the SDP relaxation for one hidden layer neural network with ReLU activation proposed by Raghunathan et al. (2018a). We prove a generalization bound regarding the surrogate adversarial loss, and show that this generalization bound does not have explicit dimension dependence if the weight matrix of the first layer has bounded ℓ_1 norm. We consider K-class classification problems in this section (i.e., $\mathcal{Y} = [K]$), and start with the definition and property of the SDP surrogate loss. Since we only have one hidden layer, $f_{\mathbf{W}}(\mathbf{x}) = \mathbf{W}_2 \rho(\mathbf{W}_1 \mathbf{x})$. Let $\mathbf{w}_{2,k}$ be the k-th column of $\mathbf{W}_2^{\top}$. Then, we have the following results according to Raghunathan et al. (2018a).

Theorem 7. *(Raghunathan et al., 2018a) For any $(\mathbf{x}, y)$, $\mathbf{W}_1$, $\mathbf{W}_2$, and $y' \neq y$,*

$$\max_{\mathbf{x}' \in \mathbb{B}_{\mathbf{x}}^{\infty}(\epsilon)} ([f_{\mathbf{W}}(\mathbf{x}')]_{y'} - [f_{\mathbf{W}}(\mathbf{x}')]_{y}) \leq [f_{\mathbf{W}}(\mathbf{x})]_{y'} - [f_{\mathbf{W}}(\mathbf{x})]_{y} + \frac{\epsilon}{4} \max_{\mathbf{P} \succeq 0, \mathrm{diag}(\mathbf{P}) \leq 1} \langle Q(\mathbf{w}_{2,y'} - \mathbf{w}_{2,y}, \mathbf{W}_1), \mathbf{P} \rangle,$$

where $Q(\mathbf{v}, \mathbf{W}) :=$

$$\begin{bmatrix} 0 & 0 & \mathbf{1}^{\top}\mathbf{W}^{\top}\mathrm{diag}(\mathbf{v}) \\ 0 & 0 & \mathbf{W}^{\top}\mathrm{diag}(\mathbf{v}) \\ \mathrm{diag}(\mathbf{v})^{\top}\mathbf{W}\mathbf{1} & \mathrm{diag}(\mathbf{v})^{\top}\mathbf{W} & 0 \end{bmatrix}. \quad (7)$$

Since we consider multi-class classification problems in this section, we use the ramp loss ϕ_{γ} defined in (4) composed with the margin operator as our loss function. Thus, we have $\ell(f_{\mathbf{W}}(\mathbf{x}), y) = \phi_{\gamma}(M(f_{\mathbf{W}}(\mathbf{x}), y))$ and $\widetilde{\ell}(f_{\mathbf{W}}(\mathbf{x}), y) = \max_{\mathbf{x}' \in \mathbb{B}_{\mathbf{x}}^{\infty}(\epsilon)} \phi_{\gamma}(M(f_{\mathbf{W}}(\mathbf{x}'), y))$. Here, we design a surrogate loss $\widehat{\ell}(f_{\mathbf{W}}(\mathbf{x}), y)$ based on Theorem 7.

Lemma 1. *Define*

$$\widehat{\ell}(f_{\mathbf{W}}(\mathbf{x}), y) := \phi_{\gamma}\Big(M(f_{\mathbf{W}}(\mathbf{x}), y) - \frac{\epsilon}{2} \max_{k \in [K], z = \pm 1} \max_{\mathbf{P} \succeq 0, \mathrm{diag}(\mathbf{P}) \leq 1} \langle z Q(\mathbf{w}_{2,k}, \mathbf{W}_1), \mathbf{P} \rangle\Big).$$

Then, we have

$$\max_{\mathbf{x}' \in \mathbb{B}_{\mathbf{x}}^{\infty}(\epsilon)} \mathbb{1}(y \neq \arg\max_{y' \in [K]} [f_{\mathbf{W}}(\mathbf{x}')]_{y'}) \leq \widehat{\ell}(f_{\mathbf{W}}(\mathbf{x}), y) \leq \mathbb{1}\Big(M(f_{\mathbf{W}}(\mathbf{x}), y) - \frac{\epsilon}{2} \max_{k \in [K], z = \pm 1} \max_{\mathbf{P} \succeq 0, \mathrm{diag}(\mathbf{P}) \leq 1} \langle z Q(\mathbf{w}_{2,k}, \mathbf{W}_1), \mathbf{P} \rangle \leq \gamma\Big).$$

We prove Lemma 1 in Appendix D.2. With this surrogate adversarial loss in hand, we can develop the following margin bound for adversarial generalization. In this theorem, we use $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n] \in \mathbb{R}^{d \times n}$, and $d_{\max} = \max\{d, d_1, K\}$.

Theorem 8 (Main Result 4). *Consider the neural network hypothesis class $\mathcal{F} = \{f_{\mathbf{W}}(\mathbf{x}) : \mathbf{W} = (\mathbf{W}_1, \mathbf{W}_2), \|\mathbf{W}_h\|_{\sigma} \leq s_h, h = 1, 2, \|\mathbf{W}_1\|_1 \leq b_1, \|\mathbf{W}_2^{\top}\|_{2,1} \leq b_2\}$. Then, for any fixed $\gamma > 0$, with probability at least $1 - \delta$, we have for all $f_{\mathbf{W}}(\cdot) \in \mathcal{F}$,*

$$\mathbb{P}_{(\mathbf{x}, y) \sim \mathcal{D}} \left\{ \exists\, \mathbf{x}' \in \mathbb{B}_{\mathbf{x}}^{\infty}(\epsilon) \ s.t.\ y \neq \arg\max_{y' \in [K]} [f_{\mathbf{W}}(\mathbf{x}')]_{y'} \right\}$$
$$\leq \frac{1}{n} \sum_{i=1}^{n} E_i + \frac{1}{\gamma} \left(\frac{4}{n^{3/2}} + \frac{60 \log(n) \log(2 d_{\max})}{n} s_1 s_2 A \right) + \frac{2 \epsilon b_1 b_2}{\gamma \sqrt{n}} + 3 \sqrt{\frac{\log \frac{2}{\delta}}{2n}},$$

where

$$E_i = \mathbb{1}\Big([f_{\mathbf{W}}(\mathbf{x}_i)]_{y_i} \leq \gamma + \max_{y' \neq y_i} [f_{\mathbf{W}}(\mathbf{x}_i)]_{y'} + \frac{\epsilon}{2} \max_{k \in [K], z = \pm 1} \max_{\mathbf{P} \succeq 0, \mathrm{diag}(\mathbf{P}) \leq 1} \langle z Q(\mathbf{w}_{2,k}, \mathbf{W}_1), \mathbf{P} \rangle \Big),$$

$$A = \Big((\frac{b_1}{s_1})^{2/3} + (\frac{b_2}{s_2})^{2/3} \Big)^{3/2} \|\mathbf{X}\|_F.$$

We prove Theorem 8 in Appendix D.3. Similar to linear classifiers, in the adversarial setting, if we have an ℓ_1 norm constraint on the matrix matrix $\mathbf{W}_1$, then the generalization bound of the surrogate adversarial loss does not have an explicit dimension dependence.

5. Experiments

In this section, we validate our theoretical findings for linear classifiers and neural networks via experiments. Our experiments are implemented with Tensorflow (Abadi et al., 2016) on the MNIST dataset (LeCun et al., 1998).

5.1. Linear Classifiers

We validate two theoretical findings for linear classifiers: (i) controlling the ℓ_1 norm of the model parameters can reduce the adversarial generalization error, and (ii) there is a dimension dependence in adversarial generalization, i.e., adversarially robust generalization is harder when the dimension of the feature space is higher. We train the multi-class linear classifier using the following objective function:

$$\min_{\mathbf{W}} \frac{1}{n}\sum_{i=1}^{n} \max_{\mathbf{x}_i'\in\mathbb{B}_{\mathbf{x}_i}^\infty(\epsilon)} \ell(f_{\mathbf{W}}(\mathbf{x}_i'), y_i) + \lambda\|\mathbf{W}\|_1, \quad (8)$$

where $\ell(\cdot)$ is cross entropy loss and $f_{\mathbf{W}}(\mathbf{x}) \equiv \mathbf{W}\mathbf{x}$. Since we focus on the generalization property, we use a small number of training data so that the generalization gap is more significant. More specifically, in each run of the training algorithm, we randomly sample $n = 1000$ data points from the training set of MNIST as the training data, and run adversarial training to minimize the objective (8). Our training algorithm alternates between mini-batch stochastic gradient descent with respect to $\mathbf{W}$ and computing adversarial examples on the chosen batch in each iteration. Here, we note that since we consider linear classifiers, the adversarial examples can be analytically computed according to Appendix C.2.

In our first experiment, we vary the values of ϵ and λ, and for each (ϵ, λ) pair, we conduct 10 runs of the training algorithm, and in each run we sample the 1000 training data independently. In Figure 2, we plot the adversarial generalization error as a function of ϵ and λ, and the error bars show the standard deviation of the 10 runs. As we can see, when λ increases, the generalization gap decreases, and thus we conclude that ℓ_1 regularization is helpful for reducing adversarial generalization error.

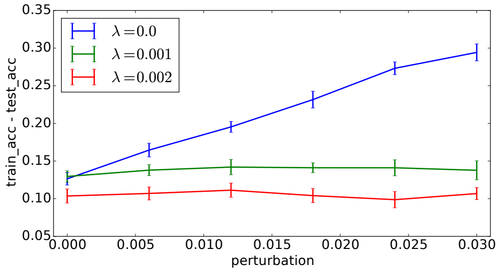

Figure 1. Linear classifiers. Adversarial generalization error vs ℓ_∞ perturbation ϵ and regularization coefficient λ.

In our second experiment, we choose $\lambda = 0$ and study the dependence of adversarial generalization error on the dimension of the feature space. Recall that each data point in the original MNIST dataset is a 28×28 image, i.e., $d = 784$. We construct two additional image datasets with $d = 196$ (downsampled) and $d = 3136$ (expanded), respectively. To construct the downsampled image, we replace each 2×2 patch—say, with pixel values a, b, c, d—on the original image with a single pixel with value $\sqrt{a^2 + b^2 + c^2 + d^2}$. To construct the expanded image, we replace each pixel—say, with value a—on the original image with a 2×2 patch, with the value of each pixel in the patch being $a/2$. Such construction keeps the ℓ_2 norm of the every single image the same across the three datasets, and thus leads a fair comparison. The adversarial generalization error is plotted in Figure 2, and as we can see, when the dimension d increases, the generalization gap also increases.

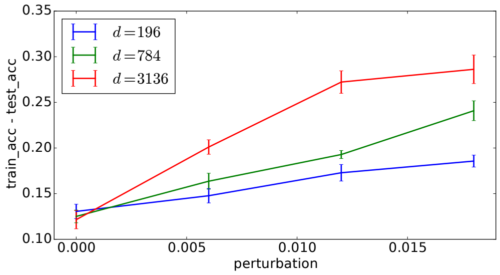

Figure 2. Linear classifiers. Adversarial generalization error vs ℓ_∞ perturbation ϵ and dimension of feature space d.

5.2. Neural Networks

In this experiment, we validate our theoretical result that ℓ_1 regularization can reduce the adversarial generalization error on a four-layer ReLU neural network, where the first two layers are convolutional and the last two layers are fully connected. We use PGD attack (Madry et al., 2017) adversarial training to minimize the ℓ_1 regularized objective (8). We use the whole training set of MNIST, and once the model is obtained, we use PGD attack to check the adversarial training and test error. We present the adversarial generalization errors under the PGD attack in Figure 3. As we can see, the adversarial generalization error decreases as we increase the regularization coefficient λ; thus ℓ_1 regularization indeed reduces the adversarial generalization error under the PGD attack.

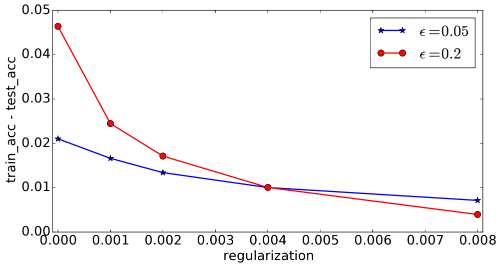

Figure 3. Neural networks. Adversarial generalization error vs regularization coefficient λ.

6. Related Work

During the preparation of the initial draft of this paper, we become aware of another independent and concurrent work by Khim & Loh (2018), which studies a similar problem. In this section, we first compare our work with Khim & Loh (2018) and then discuss other related work. We make the comparison in the following aspects.

- For binary classification problems, the adversarial

Rademacher complexity *upper bound* by Khim & Loh (2018) is similar to ours. However, we provide an adversarial Rademacher complexity *lower bound* that *matches* the upper bound. Our lower bound shows that the adversarial Rademacher complexity is never smaller than that in the natural setting, indicating the hardness of adversarially robust generalization. As mentioned, although our lower bound is for Rademacher complexity rather than generalization, Rademacher complexity is a tight bound for the rate of uniform convergence of a loss function class (Koltchinskii et al., 2006) and thus in many settings can be a tight bound for generalization. In addition, we provide a lower bound for the adversarial Rademacher complexity for neural networks. These lower bounds do not appear in the work by Khim & Loh (2018).

- We discuss the generalization bounds for the multi-class setting, whereas Khim & Loh (2018) focus only on binary classification.
- Both our work and Khim & Loh (2018) prove adversarial generalization bound using surrogate adversarial loss (upper bound for the actual adversarial loss). Khim & Loh (2018) use a method called *tree transform* whereas we use the SDP relaxation proposed by (Raghunathan et al., 2018a). These two approaches are based on different ideas and thus we believe that they are not directly comparable.

We proceed to discuss other related work.

Adversarially robust generalization As discussed in Section 1, it has been observed by Madry et al. (2017) that there might be a significant generalization gap when training deep neural networks in the adversarial setting. This generalization problem has been further studied by Schmidt et al. (2018), who show that to correctly classify two separated d-dimensional spherical Gaussian distributions, in the natural setting one only needs $\mathcal{O}(1)$ training data, but in the adversarial setting one needs $\Theta(\sqrt{d})$ data. Getting distribution agnostic generalization bounds (also known as the PAC-learning framework) for the adversarial setting is proposed as an open problem by Schmidt et al. (2018). In a subsequent work, Cullina et al. (2018) study PAC-learning guarantees for binary linear classifiers in the adversarial setting via VC-dimension, and show that the VC-dimension does not increase in the adversarial setting. This result does not provide explanation to the empirical observation that adversarially robust generalization may be hard. In fact, although VC-dimension and Rademacher complexity can both provide valid generalization bounds, VC-dimension usually depends on the number of parameters in the model while Rademacher complexity usually depends on the norms of the weight matrices and data points, and can often provide tighter generalization bounds (Bartlett, 1998). Suggala et al. (2018) discuss a similar notion of adversarial risk but do not prove explicit generalization bounds. Attias et al. (2018) prove adversarial generalization bounds in a setting where the number of potential adversarial perturbations is finite, which is a weaker notion than the ℓ_∞ attack that we consider.

Provable defense against adversarial attacks Besides generalization property, another recent line of work aim to design provable defense against adversarial attacks. Two examples of provable defense are SDP relaxation (Raghunathan et al., 2018a;b) and LP relaxation (Kolter & Wong, 2017; Wong et al., 2018). The idea of these methods is to construct *upper bounds* of the adversarial risk that can be efficiently evaluated and optimized. The analyses of these algorithms usually focus on minimizing training error and do not have generalization guarantee; in contrast, we focus on generalization property in this paper.

Generalization of neural networks Generalization of neural networks has been an important topic, even in the natural setting where there is no adversary. The key challenge is to understand why deep neural networks can generalize to unseen data despite the high capacity of the model class. The problem has received attention since the early stage of neural network study (Bartlett, 1998; Anthony & Bartlett, 1999). Recently, understanding generalization of deep nets is raised as an open problem since traditional techniques such as VC-dimension, Rademacher complexity, and algorithmic stability are observed to produce vacuous generalization bounds (Zhang et al., 2016a). Progress has been made more recently. In particular, it has been shown that when properly normalized by the margin, using Rademacher complexity or PAC-Bayesian analysis, one can obtain generalization bounds that tend to match the experimental results (Bartlett et al., 2017; Neyshabur et al., 2017; Arora et al., 2018; Golowich et al., 2017). In addition, in this paper, we show that when the weight vectors or matrices have bounded ℓ_1 norm, there is no dimension dependence in the adversarial generalization bound. This result is consistent and related to several previous works (Lee et al., 1996; Bartlett, 1998; Mei et al., 2018; Zhang et al., 2016b).

A few other lines of work have conducted theoretical analysis of adversarial examples (Wang et al., 2017; Bubeck et al., 2018; Gilmer et al., 2018b; Dohmatob, 2018; Mahloujifar et al., 2018). We provide additional discussions on related work in Appendix A.

Acknowledgements

D. Yin is partially supported by Berkeley DeepDrive Industry Consortium. K. Ramchandran is partially supported by NSF CIF award 1703678. P. Bartlett is partially supported by NSF grant IIS-1619362. The authors would like to thank Justin Gilmer for helpful discussion and anonymous reviewers for their comments. Cloud computing resources are provided by AWS Cloud Credits for Research.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. Tensorflow: a system for large-scale machine learning. In *OSDI*, volume 16, pp. 265–283, 2016.

Anthony, M. and Bartlett, P. L. *Neural network learning: Theoretical foundations*. Cambridge University Press, 1999.

Arora, S., Ge, R., Neyshabur, B., and Zhang, Y. Stronger generalization bounds for deep nets via a compression approach. *arXiv preprint arXiv:1802.05296*, 2018.

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

Attias, I., Kontorovich, A., and Mansour, Y. Improved generalization bounds for robust learning. *arXiv preprint arXiv:1810.02180*, 2018.

Bahdanau, D., Cho, K., and Bengio, Y. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

Bartlett, P. L. The sample complexity of pattern classification with neural networks: the size of the weights is more important than the size of the network. *IEEE Transactions on Information Theory*, 44(2):525–536, 1998.

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

Bartlett, P. L., Foster, D. J., and Telgarsky, M. J. Spectrally-normalized margin bounds for neural networks. In *Advances in Neural Information Processing Systems*, 2017.

Ben-Tal, A., El Ghaoui, L., and Nemirovski, A. *Robust optimization*. Princeton University Press, 2009.

Bubeck, S., Price, E., and Razenshteyn, I. Adversarial examples from computational constraints. *arXiv preprint arXiv:1805.10204*, 2018.

Carlini, N. and Wagner, D. Defensive distillation is not robust to adversarial examples. *arXiv preprint arXiv:1607.04311*, 2016.

Carlini, N. and Wagner, D. Adversarial examples are not easily detected: Bypassing ten detection methods. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pp. 3–14. ACM, 2017.

Carlini, N. and Wagner, D. Audio adversarial examples: Targeted attacks on speech-to-text. *arXiv preprint arXiv:1801.01944*, 2018.

Cullina, D., Bhagoji, A. N., and Mittal, P. PAC-learning in the presence of evasion adversaries. *arXiv preprint arXiv:1806.01471*, 2018.

Dohmatob, E. Limitations of adversarial robustness: strong no free lunch theorem. *arXiv preprint arXiv:1810.04065*, 2018.

Engstrom, L., Tsipras, D., Schmidt, L., and Madry, A. A rotation and a translation suffice: Fooling CNNs with simple transformations. *arXiv preprint arXiv:1712.02779*, 2017.

Gilmer, J., Adams, R. P., Goodfellow, I., Andersen, D., and Dahl, G. E. Motivating the rules of the game for adversarial example research. *arXiv preprint arXiv:1807.06732*, 2018a.

Gilmer, J., Metz, L., Faghri, F., Schoenholz, S. S., Raghu, M., Wattenberg, M., and Goodfellow, I. Adversarial spheres. *arXiv preprint arXiv:1801.02774*, 2018b.

Golowich, N., Rakhlin, A., and Shamir, O. Size-independent sample complexity of neural networks. *arXiv preprint arXiv:1712.06541*, 2017.

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

Graves, A., Mohamed, A.-r., and Hinton, G. Speech recognition with deep recurrent neural networks. In *International Conference on Acoustics, Speech, and Signal Processing*. IEEE, 2013.

Gu, S. and Rigazio, L. Towards deep neural network architectures robust to adversarial examples. *arXiv preprint arXiv:1412.5068*, 2014.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Computer Vision and Pattern Recognition*, 2016.

Huang, R., Xu, B., Schuurmans, D., and Szepesvári, C. Learning with a strong adversary. *arXiv preprint arXiv:1511.03034*, 2015.

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

Koltchinskii, V. et al. Local rademacher complexities and oracle inequalities in risk minimization. *The Annals of Statistics*, 34(6):2593–2656, 2006.

Kolter, J. Z. and Wong, E. Provable defenses against adversarial examples via the convex outer adversarial polytope. *arXiv preprint arXiv:1711.00851*, 2017.

Kos, J., Fischer, I., and Song, D. Adversarial examples for generative models. In *2018 IEEE Security and Privacy Workshops (SPW)*, pp. 36–42. IEEE, 2018.

Kuznetsov, V., Mohri, M., and Syed, U. Rademacher complexity margin bounds for learning with a large number of classes. In *ICML Workshop on Extreme Classification: Learning with a Very Large Number of Labels*, 2015.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Ledoux, M. and Talagrand, M. *Probability in Banach Spaces: isoperimetry and processes*. Springer Science & Business Media, 2013.

Lee, W. S., Bartlett, P. L., and Williamson, R. C. Efficient agnostic learning of neural networks with bounded fan-in. *IEEE Transactions on Information Theory*, 42(6): 2118–2132, 1996.

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

Mahloujifar, S., Diochnos, D. I., and Mahmoody, M. The curse of concentration in robust learning: Evasion and poisoning attacks from concentration of measure. *arXiv preprint arXiv:1809.03063*, 2018.

Maximov, Y. and Reshetova, D. Tight risk bounds for multi-class margin classifiers. *Pattern Recognition and Image Analysis*, 26(4):673–680, 2016.

Mei, S., Montanari, A., and Nguyen, P.-M. A mean field view of the landscape of two-layers neural networks. *arXiv preprint arXiv:1804.06561*, 2018.

Mohri, M., Rostamizadeh, A., and Talwalkar, A. *Foundations of machine learning*. MIT press, 2012.

Neyshabur, B., Bhojanapalli, S., McAllester, D., and Srebro, N. A PAC-bayesian approach to spectrally-normalized margin bounds for neural networks. *arXiv preprint arXiv:1707.09564*, 2017.

Raghunathan, A., Steinhardt, J., and Liang, P. Certified defenses against adversarial examples. *arXiv preprint arXiv:1801.09344*, 2018a.

Raghunathan, A., Steinhardt, J., and Liang, P. S. Semidefinite relaxations for certifying robustness to adversarial examples. In *Advances in Neural Information Processing Systems*, 2018b.

Schmidt, L., Santurkar, S., Tsipras, D., Talwar, K., and Madry, A. Adversarially robust generalization requires more data. *arXiv preprint arXiv:1804.11285*, 2018.

Shaham, U., Yamada, Y., and Negahban, S. Understanding adversarial training: Increasing local stability of neural nets through robust optimization. *arXiv preprint arXiv:1511.05432*, 2015.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484, 2016.

Suggala, A. S., Prasad, A., Nagarajan, V., and Ravikumar, P. On adversarial risk and training. *arXiv preprint arXiv:1806.02924*, 2018.

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

Wang, Y., Jha, S., and Chaudhuri, K. Analyzing the robustness of nearest neighbors to adversarial examples. *arXiv preprint arXiv:1706.03922*, 2017.

Wong, E., Schmidt, F., Metzen, J. H., and Kolter, J. Z. Scaling provable adversarial defenses. *arXiv preprint arXiv:1805.12514*, 2018.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016a.

Zhang, Y., Lee, J. D., and Jordan, M. I. ℓ_1-regularized neural networks are improperly learnable in polynomial time. In *International Conference on Machine Learning*, 2016b.