Robustness, Privacy, and Generalization of Adversarial Training

Fengxiang He [*†] Shaopeng Fu [*†] Bohan Wang [*‡] Dacheng Tao [†]

Abstract

Adversarial training can considerably robustify deep neural networks to resist adversarial attacks. However, some works suggested that adversarial training might comprise the privacy-preserving and generalization abilities. This paper establishes and quantifies the privacy-robustness trade-off and generalization-robustness trade-off in adversarial training from both theoretical and empirical aspects. We first define a notion, *robustified intensity* to measure the robustness of an adversarial training algorithm. This measure can be approximate empirically by an asymptotically consistent empirical estimator, *empirical robustified intensity*. Based on the robustified intensity, we prove that (1) adversarial training is (ε, δ)-differentially private, where the magnitude of the differential privacy has a positive correlation with the robustified intensity; and (2) the generalization error of adversarial training can be upper bounded by an $\mathcal{O}(\sqrt{\log N}/N)$ on-average bound and an $\mathcal{O}(1/\sqrt{N})$ high-probability bound, both of which have positive correlations with the robustified intensity. Additionally, our generalization bounds do not explicitly rely on the parameter size which would be prohibitively large in deep learning. Systematic experiments on standard datasets, CIFAR-10 and CIFAR-100, are in full agreement with our theories. The source code package is available at `https://github.com/fshp971/RPG`.

Keywords: adversarial training, adversarial robustness, privacy preservation, generalization.

1 Introduction

Adversarial training [14, 48, 3, 96] can considerably improve the adversarial robustness of deep neural networks against adversarial examples [6, 80, 26, 65, 91]. Specifically, adversarial training can be formulated as solving the following minimax-loss problem,

$$\min_{\theta} \frac{1}{N} \sum_{i=1}^{N} \max_{\|x_i' - x_i\| \leq \rho} \ell(h_\theta(x_i'), y_i),$$

where h_θ is the hypothesis parameterized by θ, N is the training sample size, x_i is a feature, y_i is the corresponding label, and l is the loss function. Intuitively, adversarial training optimizes neural networks according to the performance on worst-case examples, which are most likely to be adversarial examples.

This paper studies how adversarial training would influence the privacy-preserving [17, 18] and generalization [86, 58] abilities, both of which are of profound importance in machine learning. Based on both theoretical

[*]These authors contributed equally.

[†]F. He, S. Fu, and D. Tao are with UBTECH Sydney AI Centre, School of Computer Science, Faculty of Engineering, the University of Sydney, Darlington NSW 2008, Australia. Email: fengxiang.he@sydney.edu.au, shfu7008@uni.sydney.edu.au, and dacheng.tao@sydney.edu.au.

[‡]B. Wang was with School of Mathematical Sciences, University of Science and Technology of China, Hefei, Anhui, 230026, China, when this work was completed. He is currently with Microsoft Research Asia, Beijing, 100089, China. Email: bhwangfy@gmail.com.

and empirical evidence, we prove that:

> *The minimax-based approach can hurt the privacy-preserving and generalization abilities, while it can enhance the adversarial robustness.*

The first question raised is *how to measure adversarial robustness?* Two straightforward measures would be the accuracy on the adversarial examples and the radius ρ in adversarial training. However, it might be difficult to develop theoretical foundations upon either of them.

In this paper, we define a new term, *robustified intensity*, to assess the adversarial robustness of a learning algorithm. It is defined as the difference in the gradient norm introduced by the adversarial training, measured by the division operation. We further define an empirical estimator, *empirical robustified intensity*, for practical utilization. We prove that empirical robustified intensity is asymptotically consistent with robustified intensity. A comprehensive empirical study demonstrates that there is a clear positive correlation between robustified intensity and adversarial accuracy. This implies that robustified intensity is an informative measure.

We then study the privacy-robustness relationship. Instead of optimizing the average performance on all training examples, adversarial training optimizes neural networks on worst-case examples. This forces the learned model more heavily relying on a small subset of the training sample set. Therefore, one may have a considerably increased chance to launch a successful *differential attack*, which first replaces one training example by a fake example, and then inference other training examples by the change of the output model. We prove that adversarial training is (ε, δ)-differentially private when using stochastic gradient descent (SGD) to optimize the minimax loss. Further, the magnitude of both ε and δ have a positive correlation with the robustified intensity, which is the first result that establishes the theoretical foundations for the privacy-robustness trade-off.

Based on the privacy preservation, we prove an $\mathcal{O}(\sqrt{\log N}/N)$ on-average generalization bound and an $\mathcal{O}(1/\sqrt{N})$ high-probability generalization bound for adversarial training, where N is the training sample size. The two bounds are established based on a novel theorem linking algorithmic stability and differential privacy. Furthermore, our generalization bounds do not have any explicit dependence on the parameter size, which can be prohibitively large in deep learning. The only term that would rely on the model size, the norm of the gradient, is verified by the experiments to be small.

From the empirical aspect, we conduct systematic experiments on two standard datasets, CIFAR-10 and CIFAR-100 [41], with two different metric norms, L_∞ and L_2, for adversarial training while strictly controlling irrelative variables. The privacy-preserving abilities are measured by the accuracies of the membership inference attack [73, 93]. Meanwhile, the generalizabilities are measured by the difference between the training accuracies and the test accuracies. The membership inference attack accuracies, generalization gaps, and empirical robustified intensities of the models trained in various settings are collected for analysis. The empirical results are in full agreement with our hypotheses. The training code, learned models, and collected data are available at `https://github.com/fshp971/RPG`.

The rest of this paper is organized as follows. Section 2 reviews related works. Section 3 presents notations and preliminaries necessary to the following discussions. Section 4 defines the robustified intensity and its empirical estimator. Sections 5 and 6 establish the privacy-robustness relationship and generalization-robustness relationship, respectively. Section 7 collects all the omitted proofs. Section 8 presents implementation details of our experiments. Section 9 concludes this paper.

2 Background

This section reviews the background of this work.

Deep learning theory. Deep learning has been deployed successfully in many real-world scenarios. However, the theoretical foundations of deep learning are still elusive. For example, there is no explanation for how deep learning algorithms work, why they can succeed, when they would fail, and whether they would hurt society. Such deficiency in explainability questions the transparency and accountability of deep learning, and further undermines our confidence of deploying deep learning in security-critical application domains, such [51, 79], medical diagnosis [42, 74], and drug discovery [11]. Many works have emerged to establish the theoretical foundations of deep learning via VC dimension [28], Rademacher complexity [25, 4], covering number [4], Fisher-Rao norm [49, 83], PAC-Bayesian framework [62], algorithmic stability [27, 45, 87], and the dynamics of stochastic gradient descent or its variants [54, 59, 30]. Please see more related works in surveys [20, 31, 68]. This work is committed to establishing theoretical foundations of privacy, generalization, adversarial attack in deep learning, all of which have profound importance in enhancing the explainability, transparency, and accountability of deep models.

Generalization. Good generalization guarantees that an algorithm learns the underlying patterns in training data rather than just memorize the data. In this way, good generalization abilities provide confidence that the models trained on existing data can be applied to similar but unseen scenarios. Three major approaches in analyzing the generalizability are seen in the literature: (1) generalization bounds based on the hypothesis complexity, including VC dimension [7, 85], Rademacher complexity [40, 39, 5], and covering number [15, 29]. The results are usually obtained via concentration inequalities. They also suggest controlling the model size to secure the generalizability, which is no longer valid in deep learning; (2) generalization bounds based on the algorithmic stability [70, 8, 92]. The results in this stream follow the motivation that learning algorithms robust to small disturbances in input data usually have good generalizability; and (3) generalization bounds in the PAC-Bayes framework [55, 56]. The results are obtained based on information-theoretical versions of concentration inequalities.

Privacy preservation. Deep learning has become a dominant player in many real-world application areas, including financial services [22], healthcare [88], and biometric authentication [76], where massive personal data has been collected. However, an increasing number of privacy breaches have been reported. An infamous scandal in 2018 shocked people that Cambridge Analytics harvested large amounts of personal data without consent for political advertising. The customers are fed meticulously selected advertisement to promote specific politicians and agendas. It sheds lights to the prohibitive reality that machine learning algorithms can quietly navigate consumers' choice by the data that was supposed to be private.

A popular measure for the privacy-preserving ability is differential privacy [18] based on the privacy loss; please see for more details. The magnitude of differential privacy (ε, δ) represents the ability to resist *differential attacks* that employing fake data to steal private information in the training data. Many variants of differential privacy have emerged to date: (1) concentration differential privacy is proposed under assumptions that the distribution of the privacy loss is sub-Gaussian [19, 9]; (2) mutual-information differential privacy replace the division operation in difference privacy by mutual information [13, 90, 50]; (3) KL differential privacy replace the mutual information by the KL divergence [10]; (4) similarly, Rényi differential privacy replaces the KL divergence by Rényi divergence further [57, 23]. Abadi et al. [1] and Arachchige et al. [2] have also studied the privacy preservation of deep learning.

Adversarial robustness. Many works suggest that adversarial examples are widespread in the feature spaces of deep models [6, 80, 26, 65, 12, 35]. Specifically, for (almost) any training example, one can find an adversarial example closed to it but the neural network assigns the adversarial example to a different class.

Thus, one can slightly modify an example to fool a neural network. This would expose deep learning-based systems to adversarial attacks [63, 43, 53, 64, 24]. Many defence strategies [14, 48, 3, 96, 89] can increase the robustness of neural networks to adversarial attacks [63, 43, 53, 64, 24]. However, it is also reported that these approaches would undermine privacy preservation and generalization.

Generalization-robustness relationship. Some works have studied the trade-off between generalization and robustness. Tsipras *et al.* [82] prove the existence of a trade-off between the standard accuracy of a model and its robustness to adversarial perturbations. Sun *et al.* [78] prove that adversarial training needs more training data to achieve the same test accuracy as standard ERM. Nakkiran [60] suggest that "robust classification may require more complex classifiers (*i.e.*, more capacity) than standard classification". Additionally, they prove a quantitative trade-off between the robustness and standard accuracy for simple classifiers. Three $\mathcal{O}(1/\sqrt{N})$ generalization bounds are given by [94, 37, 84], which are based on the Rademacher complexity and covering number of the hypothesis space. A detailed comparison of the tightness is given in Section 6. Schmidt *et al.* [71] prove that the hypothesis complexity of models learned by adversarial training is larger than those learned by empirical risk minimization (ERM), which is also verified empirically. However, the existing results relying on hypothesis capacity/complexity of neural networks, which are prohibitively large. Our paper proposes two novel generalization bounds at rate $\mathcal{O}(\sqrt{\log N}/N)$ and $\mathcal{O}(1/\sqrt{N})$, respectively, without explicitly relying on the capacity/complexity. Instead, gradient norm, the only factor in our bounds that could depend on the parameter size, is verified to be considerably small by experiments.

Robustness-privacy relationship. There have been initial attempts to study the robustness-privacy relationship. Some works suggest that differentially private machine learning algorithms are robust to adversarial examples [46, 47]. Pinot *et al.* [67] define two terms, adversarial robustness and generalized adversarial robustness, to express the robustness to adversarial examples, which are similar to the differential privacy and its variants. The paper then argues that the two new terms are equivalent to Rényi differential privacy, but without theoretical proof. Phan *et al.* [66] design algorithms with both theoretical guarantees in the privacy-preserving ability and adversarial robustness. Song *et al.* [77] conduct comprehensive experiments to investigate the relationship between robustness and privacy, with the results suggesting that adversarial training has privacy risks. However, there is so far no theoretical foundation has been established to discover the robustness-privacy relationship.

3 Notations

Suppose $S = \{(x_1, y_1), \ldots, (x_N, y_N) | x_i \in \mathbb{R}^{d_X}, y_i \in \mathbb{R}^{d_Y}, i = 1, \ldots, N\}$ is a sample set, where d_X and d_Y are the dimension of the feature X and the label Y, respectively. For the brevity, we define $z_i = (x_i, y_i)$, which is an independent and identically distributed (i.i.d.) observation of variable $Z = (X, Y) \in \mathcal{Z}$.

Differential privacy measures the ability of preserving privacy [18]. A stochastic algorithm $\mathcal{A}$ is called (ε, δ)-differentially private, if for any subset $B \subset \mathcal{H}$ and any neighboring sample set pair S and S' which are different by only one example, we have

$$\log \left[\frac{\mathbb{P}_{\mathcal{A}(S)}(\mathcal{A}(S) \in B) - \delta}{\mathbb{P}_{\mathcal{A}(S')}(\mathcal{A}(S') \in B)} \right] \leq \varepsilon. \tag{1}$$

Algorithm $\mathcal{A}$ is also called ε-differentially private, if it is $(\varepsilon, 0)$-differentially private. Differential privacy controls the privacy loss (cf. [18], p.18) defined as below,

$$\log \left[\frac{\mathbb{P}_{\mathcal{A}(S)}(\mathcal{A}(S) \in B)}{\mathbb{P}_{\mathcal{A}(S')}(\mathcal{A}(S') \in B)} \right].$$

For the hypothesis $\mathcal{A}(S)$ learned by an algorithm $\mathcal{A}$ on the training sample set S, the expected risk $\mathcal{R}(\mathcal{A}(S))$ and empirical risk $\hat{\mathcal{R}}(\mathcal{A}(S))$ of the algorithm $\mathcal{A}$ are defined as follows,

$$\mathcal{R}(\mathcal{A}(S)) = \mathbb{E}_Z \ell(\mathcal{A}(S), Z),$$
$$\hat{\mathcal{R}}_S(\mathcal{A}(S)) = \frac{1}{N}\sum_{i=1}^{N} \ell(\mathcal{A}(S), z_i).$$

It worths noting that the randomness of $\mathcal{A}(S)$ can come from both the stochastic algorithm $\mathcal{A}$ and the training sample set S. Then, the generalization error is defined as the difference between the expected risk and empirical risk, whose upper bound is called the generalization bound.

Learning algorithms usually solve the following empirical risk minimization (ERM) problem to approach the optimal hypothesis,

$$\min_{\theta} \hat{\mathcal{R}}_S(\theta) = \min_{\theta} \frac{1}{N}\sum_{i=1}^{N} \ell(h_\theta(x_i), y_i).$$

We usually employ stochastic gradient-based optimizers for ERM in deep learning. Popular options of stochastic gradient-based optimizers include stochastic gradient descent (SGD) [69], momentum [61, 81], and Adam [38]. For the brevity, we analyze SGD in this paper. The analysis for other stochastic gradient-based optimizers is similar.

Suppose $\mathcal{B}$ is a mini batch randomly drawn from the training sample set S. Then, the stochastic gradient on $\mathcal{B}$ is as follows,

$$\hat{g}^{\mathrm{ERM}}(\theta) = \frac{1}{|\mathcal{B}|}\sum_{(x_i,y_i)\in\mathcal{B}} \nabla_\theta \ell(h_\theta(x_i), y_i).$$

In the t-th iteration, the weight is updated as follows,

$$\theta_{t+1}^{\mathrm{ERM}} = \theta_t^{\mathrm{ERM}} - \eta_t \hat{g}^{\mathrm{ERM}}(\theta_t^{\mathrm{ERM}}),$$

where θ_t^{ERM} is the weight vector in the t-th iteration and η_t is the corresponding learning rate.

Meanwhile, adversarial training employs SGD to solve the following minimax problem,

$$\min_{\theta} \hat{\mathcal{R}}_S^A(\theta) = \min_{\theta} \frac{1}{N}\sum_{i=1}^{N} \max_{\|x_i'-x_i\|\le\rho} \ell(h_\theta(x_i'), y_i), \tag{2}$$

where ρ is the radius of the ball centered at the example (x_i, y_i). Here, we call $\hat{\mathcal{R}}_S^A(\theta)$ adversarial empirical risk. Correspondingly, the stochastic gradient on a mini batch $\mathcal{B}$ and the weight update are calculated as below,

$$\hat{g}^A(\theta) = \frac{1}{|\mathcal{B}|}\sum_{(x_i,y_i)\in\mathcal{B}} \nabla_\theta \max_{\|x_i'-x_i\|\le\rho} \ell(h_\theta(x_i'), y_i),$$
$$\theta_{t+1}^A = \theta_t^A - \eta_t \hat{g}^A(\theta_t^A). \tag{3}$$

4 Measurement of robustness

This section presents a new term, *robustified intensity*, to measure the adversarial robustness. We also design an asymptotically consistent empirical estimator, *empirical robustified intensity*, to empirically approximate robustified intensity.

4.1 Robustified intensity

We first define *single-iteration robustified intensity* as follows,

Definition 4.1 (Single-iteration robustified intensity)**.** *The single-iteration robustified intensity of the t-th iteration in adversarial training (eq. 2) is defined to be*

$$I_t = \frac{\max_{(x,y)\in\mathcal{Z}} \left\| \nabla_\theta \max_{\|x'-x\|\le\rho} \ell(h_\theta(x'),y)\big|_{\theta=\theta_t^A} \right\|}{\max_{(x,y)\in\mathcal{Z}} \left\| \nabla_\theta \ell(h_\theta(x),y)|_{\theta=\theta_t^{\mathrm{ERM}}} \right\|}, \tag{4}$$

where x' is arbitrary in the input space subject to the condition $\|x'-x\|\le\rho$, $\|\cdot\|$ is the norm on the parameter space, and θ_t^A and θ_t^{ERM} are the parameters in the t-th iteration of adversarial training and ERM, respectively.

For brevity, we term the nominator and the denominator as L_t^A and L_t^{ERM}, respectively. Thus, $I_t = \frac{L_t^A}{L_t^{\mathrm{ERM}}}$. In adversarial training, L_t^A is usually larger than L_t^{ERM} and thus $I_t > 1$.

We then extend our measure for adversarial robustness to whole training procedures. Suppose there are T iterations in an adversarial training process, and the corresponding single-iteration robustified intensities are $I_1, \cdots, I_T$, respectively. We define a *robustified intensity* for the whole algorithm based on the single-iteration robustified intensities as below.

Definition 4.2 (Robustified intensity)**.** *Suppose T iterations exist in an adversarial training procedure. Then, the robustified intensity for this procedure is defined to be*

$$I_{1:T} = \left(\frac{1}{T} \sum_{t=1}^{T} I_t^4 \right)^{\frac{1}{4}},$$

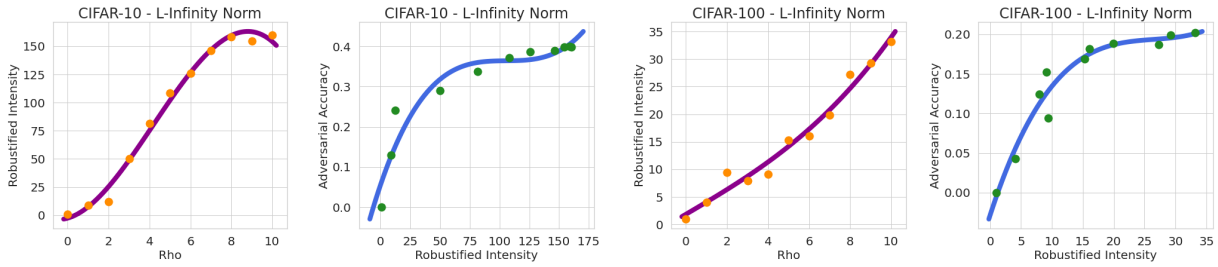
where I_t is the single-iteration robustified intensity of the t-th interation.

Remark 4.1. *The fourth-order exponential operations in Definition 4.2 are designed for the convenience in establishing the privacy-robustness relationship. Moreover, our experiments show that such designing may not compromise the efficiency in evaluating the adversarial robustness.*

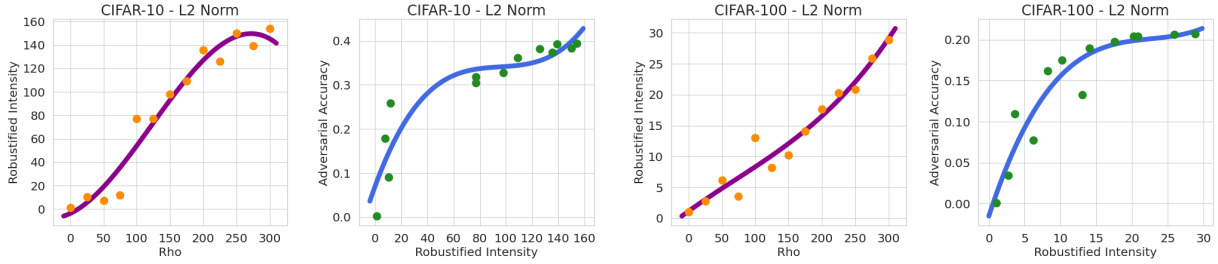
4.2 How to empirically estimate robustified intensity?

Calculating the robustified intensity is technically impossible as it involves searching the maximal value of gradient norms across the Euclidean space. We thus define *empirical single-iteration robustified intensity* and *empirical robustified intensity* to empirically estimate their theoretical counterparts.

Definition 4.3 (Empirical single-iteration robustified intensity)**.** *The empirical single-iteration robustified*



(a) CIFAR-10, L_∞ norm. (b) CIFAR-100, L_∞ norm.



(c) CIFAR-10, L_2 norm. (d) CIFAR-100, L_2 norm.

Figure 1: We conduct experiments on CIFAR-10 and CIFAR-100 with two different adversarial training metric norms, L_∞ norm and L_2 norm. For each experiment, we draw two plots: (1) robustified intensity vs. radius ρ; and (2) adversarial accuracy vs. robustified intensity. Fourth-order polynomial regression is used for curve fitting in each figure. robustified intensity has positive correlations with both radius ρ and adversarial accuracy, which demonstrates that robustified intensity is an informative robustness measurement.

intensity of the t-th iteration in adversarial training (eq. 2) is defined to be

$$\hat{I}_t = \frac{\max_{(x_i,y_i)\in\mathcal{B}} \left\| \nabla_\theta \max_{\|x_i'-x_i\|\le\rho} \ell(h_\theta(x_i'),y_i)\big|_{\theta=\theta_t^A} \right\|}{\max_{(x_i,y_i)\in\mathcal{B}} \left\| \nabla_\theta \ell(h_\theta(x_i),y_i)\big|_{\theta=\theta_t^{\mathrm{ERM}}} \right\|},$$

where $\mathcal{B}$ is a mini batch sub-sampled from the training sample set S and $\|\cdot\|$ is a norm defined in the space of the gradient.

Definition 4.4 (Empirical robustified intensity)**.** *Suppose T iterations exist in an adversarial training procedure. Then, the empirical robustified intensity for this procedure is defined to be*

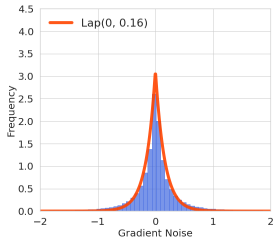
$$\hat{I}_{1:T} = \left(\frac{1}{T} \sum_{t=1}^{T} \hat{I}_t^4 \right)^{\frac{1}{4}},$$

where $\hat{I}_t$ is the empirical single-iteration robustified intensity of the t-th interation.

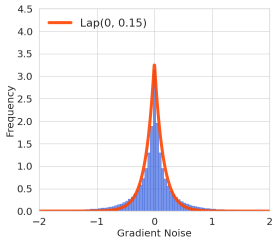
We can prove that both empirical estimators asymptotically consistent with their theoretical counterparts. The proofs need one mild assumption as below.

Assumption 4.1. *The gradient of loss function $\nabla_\theta \ell(\theta, z) \in C^0(\mathcal{Z})$; i.e., for any hypothesis $h_\theta \in \mathcal{H}$, $\nabla_\theta \ell(h_\theta, z)$ is continuous with respect to example z.*

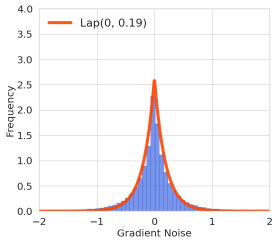
The continuity with respect to the data z can be easily meet in practice. Then, the asymptotical consistency is presented in the following two theorems.



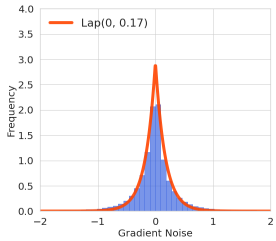
(a) CIFAR-10, $40,000$ iterations, ERM, Lap$(0, 0.16)$.



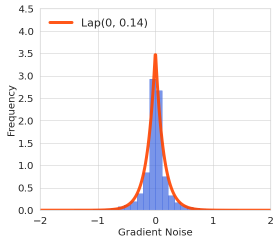
(b) CIFAR-10, $80,000$ iterations, ERM, Lap$(0, 0.15)$.



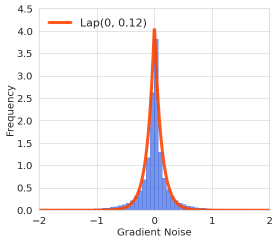
(c) CIFAR-100, $40,000$ iterations, ERM, Lap$(0, 0.19)$.



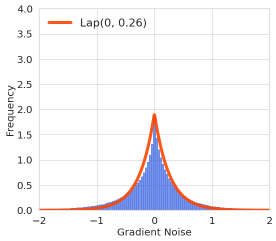
(d) CIFAR-100, $80,000$ iterations, ERM, Lap$(0, 0.17)$.



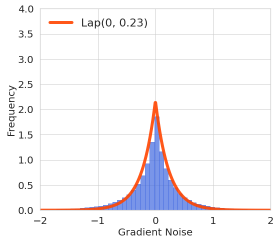
(e) CIFAR-10, $80,000$ iterations, L_∞ norm, $\rho = \frac{5}{255}$, Lap$(0, 0.14)$.



(f) CIFAR-10, $80,000$ iterations, L_2 norm, $\rho = \frac{150}{255}$, Lap$(0, 0.12)$.



(g) CIFAR-100, $80,000$ iterations, L_∞ norm, $\rho = \frac{5}{255}$, Lap$(0, 0.26)$.



(h) CIFAR-100, $80,000$ iterations, L_2 norm, $\rho = \frac{150}{255}$, Lap$(0, 0.23)$.

Figure 2: Histograms of the gradient noises from different iterations in ERM under different experimental settings. Each plot is based on $10,000,000$ gradient noises. All the plots show that the distribution of gradient noise is similar to Laplacian distribution, which justifies that it is favorable to model gradient noise with Laplacian distribution.

Theorem 4.1 (Asymptotic consistency of empirical single-iteration robustified intensity)**.** *Suppose the empirical single-iteration robustified intensity in the t-th training iteration is $\hat{I}_t^{(\tau)}$ where the batch size is τ. Then, the $\hat{I}_t^{(\tau)}$ is an unbiased estimator for the single-iteration robustified intensity I_t; i.e., $\lim_{\tau\to\infty} \hat{I}_t^{(\tau)} = I_t$.*

Theorem 4.2 (Asymptotic consistency of empirical robustified intensity)**.** *Suppose the empirical robustified intensity in the t-th training iteration is $\hat{I}_{1:T}^{(\tau)}$ where the batch size is τ. Then, the $\hat{I}_{1:T}^{(\tau)}$ is an unbiased estimator for the robustified intensity I_t; i.e., $\lim_{\tau\to\infty} \hat{I}_{1:T}^{(\tau)} = I_{1:T}$.*

These two theorems secure that when the batch size for estimation is sufficiently large, the empirical single-iteration robustified intensity rigorously equals to the single-iteration robustified intensity. The proofs are novel and technically non-trivial. Please see details in Section 7.1.

4.3 Is robustified intensity informative?

A comprehensive empirical study is conducted, comparing empirical robustified intensity $\hat{I}_{1:T}$, radius ρ, and adversarial accuracy on the CIFAR-10 and CIFAR-100 datasets. Implementation details are given in Section 8.

The empirical robustified intensity, radius, and adversarial accuracy are collected in every setting, as shown in fig. 1. Fourth-order polynomial regression is employed for curve fitting.[1] From the collected data,

[1]We grid searched different values of the polynomial regression order in $\{1, 2, \ldots, 6\}$. 4 is the "sweet point" between "under-fitting" and "over-fitting". In the rest of this paper, we employ polynomial regression for curving fitting multiple times. The orders

we obtain two major observations: (1) the (empirical) robustified intensity has a clear positive correlation with the radius ρ; and (2) the adversarial accuracy has a clear positive correlation with the robustified intensity is observed in the full interval of robustified intensity. The two observations verify that the robustified intensity is comparably informative to two standard measures for adversarial robustness, radius and adversarial accuracy.

5 Privacy-robustness trade-off

This section studies the relationship between privacy preservation and robustness in adversarial training.

5.1 What is the distribution of gradient noise?

Stochastic gradient-based optimizers introduce noise in optimization. It is interesting to ask *what is the distribution of gradient noise?* Some works [44, 52, 54, 34, 75] assumed that the gradient noise is drawn from a Gauss distribution. In this work, we conducted a large-scale experiment to investigate the distribution of gradient noise. The experiment results on CIFAR-10 and CIFAR-100 are collected in fig. 2, which demonstrate that Laplacian distribution is appropriate to model the distribution of gradient noise. Then, we make the following assumption. For more implementation details, please see Section 8.

Assumption 5.1. *The gradient calculated from a mini-batch is drawn from a Laplacian distribution centered at the empirical risk,*

$$\frac{1}{\tau}\sum_{(x,y)\in\mathcal{B}}\nabla_\theta \max_{\|x'-x\|\leq\rho}\ell(h_\theta(x'),y)\sim \mathrm{Lap}\left(\nabla_\theta\hat{\mathcal{R}}_S^A(\theta),b\right).$$

5.2 Theoretical evidence

Following the Laplacian mechanism, we approximate the differential privacy of adversarial training as the following two theorems.

Theorem 5.1 (Privacy-robustness relationship I)**.** *Suppose one employs SGD for adversarial training and the whole training procedure has T iterations. Then, the adversarial training is (ε_A,δ_A)-differentially private, where*

$$\varepsilon_A=\sqrt{2\log\frac{N}{\delta'}\sum_{t=1}^{T}\varepsilon_t^2}+\sum_{t=1}^{T}\varepsilon_t\frac{e^{\varepsilon_t}-1}{e^{\varepsilon_t}+1},$$
$$\delta_A=\frac{\delta'}{N},$$

in which

$$\varepsilon_t=\frac{2L_t^{\mathrm{ERM}}}{Nb}I_t,$$

and δ' is a positive real, I_t is the single-iteration robustified intensity in the t-th iteration, and b is the Laplacian parameter.

are selected similarly.

Theorem 5.2 (Privacy-robustness relationship II)**.** *Suppose a T-iteration SGD is employed to solve the minimax optimization problem in adversarial training. Then, the adversarial training is $(\varepsilon_A, \delta_A)$-differentially private, where*

$$\varepsilon_A = \varepsilon_{1:T}\sqrt{2T\log\frac{N}{\delta'}} + \mathcal{O}\left(\frac{1}{N^2}\right),$$
$$\delta_A = \frac{\delta'}{N},$$

where

$$\varepsilon_{1:T} = \frac{2L_{1:T}^{\mathrm{ERM}}}{Nb} I_{1:T},$$

and $L_{1:T}^{\mathrm{ERM}} := \left(\frac{1}{T}\sum_{t=1}^{T}\left(L_t^{\mathrm{ERM}}\right)^4\right)^{\frac{1}{4}}$, δ' is a positive real, $I_{1:T}$ is the robustified intensity for the whole training procedure (see Definition 4.2), and b is the Laplacian parameter.

Remark 5.1. *Theorems 5.1 and 5.2 suggest a negative correlation between adversarial robustness and privacy preservation.*

Remark 5.2. *Theorem 5.1 approximate the differential privacy of a learning algorithm based on the adversarial robustness of its every iteration ($I_1, \ldots, I_t$), while Theorem 5.2 approximate the differential privacy based on the adversarial robustness of the whole algorithm ($I_{1:T}$).*

Remark 5.3. *The approximation of differential privacy given by Theorem 5.2 is $(\mathcal{O}(\sqrt{\log N}/N), \mathcal{O}(1/N))$.*

Similarly, we can approximate the differential privacy of ERM in the following corollary.

Corollary 5.1. *Suppose one employs SGD for ERM and the whole training procedure has T iterations. Then, the ERM is (ε, δ)-differentially private, where*

$$\varepsilon = \varepsilon_{1:T}^{\mathrm{ERM}}\sqrt{2T\log\frac{N}{\delta'}} + \mathcal{O}\left(\frac{1}{N^2}\right),$$
$$\delta = \frac{\delta'}{N},$$

in which,

$$\varepsilon_{1:T}^{\mathrm{ERM}} = \frac{2L_{1:T}^{\mathrm{ERM}}}{Nb},$$

and $L_{1:T}^{\mathrm{ERM}} := \left(\frac{1}{T}\sum_{t=1}^{T}\left(L_t^{\mathrm{ERM}}\right)^4\right)^{\frac{1}{4}}$, δ' is a positive real.

Comparing the results for adversarial training and ERM, we have

$$\varepsilon_{1:T} = I_{1:T} \cdot \varepsilon_{1:T}^{\mathrm{ERM}} + \mathcal{O}(1/N^2).$$

Theorem 5.2 and Corollary 5.1 show that both factors ε and δ have positive correlations with the robustified intensity, which suggests a trade-off between privacy preservation and adversarial robustness.

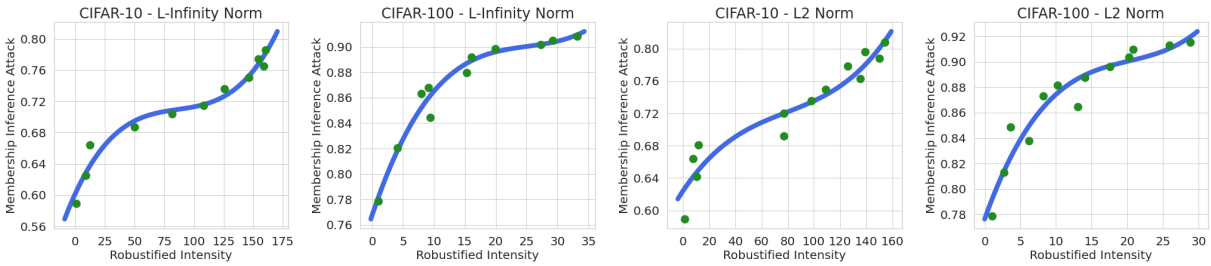


Figure 3: Plots of membership inference attack accuracy vs. empirical robustified intensity. The datasets and adversarial training metric norms of the four plots are respectively (1) CIFAR-10 and L_∞ norm; (2) CIFAR-100 and L_∞ norm; (3) CIFAR-10 and L_2 norm; and (4) CIFAR-100 and L_2 norm. Fourth-order polynomial regression is used for curve fitting in each figure. Based on these figures, we find that membership inference attack accuracy has a positive correlation with robustified intensity, which demonstrates a negative correlation between privacy preservation and adversarial robustness.

5.2.1 Proof skeleton

This section presents the proof skeleton. Detailed proofs for Theorems 5.1 and 5.2 are given in Sections 7.2 and 7.3, respectively.

It is usually hard to calculate the differential privacy of iterative algorithm directly. However, the differential privacy of every single step in the iterative algorithm can be easily derived. Based on the differential privacy of every steps, composition theorems can then approximate the differential privacy of the whole learning algorithm. The proofs for calculating the differential privacy of adversarial training follows the intuition above.

In this paper, we employs the tightest composition theorem by He *et al.* [32] as follows.

Lemma 5.1 (Advanced composition; cf. [32], Theorem 5)**.** *Suppose an iterative algorithm has T steps, and the t-th step is ε_t-differentially private. Then, the whole algorithm is (ε, δ)-differentially private, where*

$$\varepsilon = \sqrt{2 \log \frac{1}{\delta} \sum_{t=1}^{T} \varepsilon_t^2} + \sum_{t=1}^{T} \varepsilon_t \frac{e^{\varepsilon_t} - 1}{e^{\varepsilon_t} + 1},$$

and δ is a positive real.

5.3 Empirical evidence

We trained Wide ResNet on datasets CIFAR-10 and CIFAR-100 to verify the privacy-robustness trade-off. For more implementation details, please see Section 8.

The privacy-preserving ability is measured by membership inference attack [73, 93], a standard privacy attack tool. Membership inference attack aims to inference whether a given data point comes from the training set based on the output of the model. A higher membership inference attack accuracy means that a privacy attack for private information is more likely to succeed and thus implies a worse privacy-preserving ability.

The membership inference attack accuracies and empirical robustified intensities of all models are collected, as shown in fig. 3. From the four figures, we observe a clear positive correlation between the membership inference attack accuracy and the robustified intensity $I_{1:T}$, which demonstrates a negative correlation

between privacy preservation and robustness.

6 Generalization-robustness trade-off

This section studies the relationship between generalization and robustness.

We first prove a high-probability generalization bound for an (ε, δ)-differentially private machine learning algorithm. Combining the established privacy-robustness relationship, this bound helps study the generalizability of adversarial training.

Theorem 6.1 (High-probability generalization bound via differential privacy)**.** *Suppose all conditions of Theorem 5.2 hold. Then, the algorithm $\mathcal{A}$ has a high-probability generalization bound as follows. Specifically, the following inequality holds with probability at least $1-\gamma$:*

$$\mathbb{E}_{\mathcal{A}}\mathcal{R}(\mathcal{A}(S)) - \mathbb{E}_{\mathcal{A}}\hat{\mathcal{R}}_S(\mathcal{A}(S)) \leq c\left(M(1-e^{-\varepsilon}+e^{-\varepsilon}\delta)\log N \log\frac{N}{\gamma} + \sqrt{\frac{\log 1/\gamma}{N}}\right), \tag{5}$$

where γ is an arbitrary probability mass, M is the bound for loss l, N is the training sample size, c is a universal constant for any sample distribution, and the probability is defined over the sample set S.

We also prove an on-average generalization bound, which expresses the "expected" generalizability. It worths noting that high-probability generalization bounds can also lead to on-average bounds by integration in theory. However, the calculations would be prohibitively difficult. Thus, we practice an independent approach to prove the on-average bound.

Theorem 6.2 (On-average generalization bound via differential privacy)**.** *Suppose all conditions of Theorem 5.2 hold. Then, the on-average generalization error of the algorithm $\mathcal{A}$ is upper bounded by*

$$\mathbb{E}_{S,\mathcal{A}}\left[\mathcal{R}(\mathcal{A}(S)) - \hat{\mathcal{R}}_S(\mathcal{A}(S))\right] \leq M\delta e^{-\varepsilon} + M(1-e^{-\varepsilon}).$$

Remark 6.1. *By the Post-processing property of differential privacy, since $\mathcal{B}$: $h \to \max_{x'\in\mathbb{B}_*(\rho)}\ell(h,(x',*))$ is a one-to-one mapping, $\max_{x'\in\mathbb{B}_*(\rho)}\ell(\mathcal{A},(x',*))$ is (ε,δ) differentially private. Therefore, Theorem 6.1 and 6.2 hold for adversarial learning algorithms.*

Both generalization bounds have positive correlations with the magnitude of the differential privacy, which further has a positive correlation with the adversarial robustness. This leads to the following corollary.

Corollary 6.1. *There is a trade-off between generalizability and adversarial robustness (measured by robustified intensity) in adversarial training.*

6.1 Establishing generalization bounds based on algorithmic stability

Theorems 6.1 and 6.2 are established via algorithmic stability which measures how stable an algorithm is when the training sample is exposed to disturbance [70, 36, 8]. While algorithmic stability has many different definitions, this paper mainly discusses the uniform stability.

Definition 6.1 (Uniform stability; cf. [8])**.** *A machine learning algorithm $\mathcal{A}$ is uniformly stable, if for any neighboring sample pair S and S' which are different by only one example, we have the following inequality,*

$$\left|\mathbb{E}_{\mathcal{A}}\ell(\mathcal{A}(S),Z) - \mathbb{E}_{\mathcal{A}}\ell(\mathcal{A}(S'),Z)\right| \leq \beta,$$

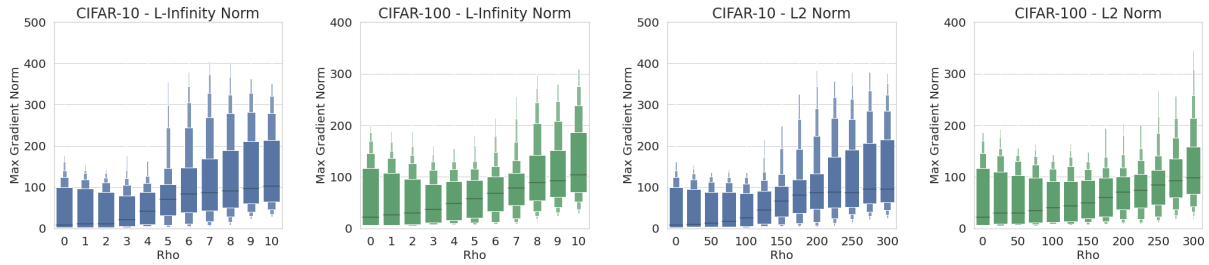


Figure 4: Box plots of the obtained L_t^{ERM} in ERM and L_t^A in adversarial training on CIFAR-10 and CIFAR-100 with different radius ρ and different adversarial training metric norms. In every setting, the training procedure has $800,000$ iterations, and the max gradient norms are calculated every 100 iterations. Therefore, the sample sizes are 800 in all settings. From the figures, we could find that the max gradient norms are small comparing to the sizes of hypothesis parameters, which verifies the tightness of our generalization bounds.

where Z is an arbitrary example, $\mathcal{A}(S)$ and $\mathcal{A}(S')$ are the output hypotheses learned on the training sets S and S', respectively, and β is a positive real constant. The constant β is called the uniform stability of the algorithm $\mathcal{A}$.

In this paper, we prove that (ε, δ)-differentially private machine learning algorithms are algorithmic stable as the following lemma.

Lemma 6.1 (Stability-privacy relationship)**.** *Suppose that a machine learning algorithm $\mathcal{A}$ is (ε, δ)-differentially private. Assume the loss function l is upper bounded by a positive real constant $M > 0$. Then, the algorithm $\mathcal{A}$ is uniformly stable,*

$$\left|\mathbb{E}_{\mathcal{A}}\ell(\mathcal{A}(S), Z) - \mathbb{E}_{\mathcal{A}}\ell(\mathcal{A}(S'), Z)\right| \leq M\delta e^{-\varepsilon} + M(1 - e^{-\varepsilon}).$$

6.1.1 Establishing high-probability generalization bound

Our high-probability generalization bound relies on the following lemma by Feldman *et al.* [21].

Lemma 6.2 (cf. [21], Theorem 1)**.** *Suppose a deterministic machine learning algorithm $\mathcal{A}$ is stable with uniform stability β. Suppose $l \leq 1$. Then, for any sample distribution and any $\gamma \in (0, 1)$, there exists a universal constant c, such that, with probability at least $1 - \gamma$ over the draw of sample, the generalization error can be upper bounded as follows,*

$$\mathbb{E}_{z\sim P}\ell(\mathcal{A}(S), z) - \frac{1}{N}\sum_{z\in S}\ell(\mathcal{A}(S), z) \leq c\left(\beta \log N log\frac{N}{\gamma} + \sqrt{\frac{\log 1/\gamma}{N}}\right).$$

Proof of Theorem 6.1. Combining Lemma 6.1 and Lemma 6.2, we can directly prove Theorem 6.1. □

6.1.2 Establishing on-average generalization bound

Our on-average generalization bound relies on the following lemma by Dwork *et al.* [16].

Lemma 6.3 (Lemma 11, cf. [72])**.** *Suppose the loss function is upper bounded. For any machine learning algorithm with β Replace-one stability, its generalization error is upper bound as follows,*

$$\mathcal{R}(\mathcal{A}(S)) - \hat{\mathcal{R}}_S(\mathcal{A}(S)) \leq \beta.$$

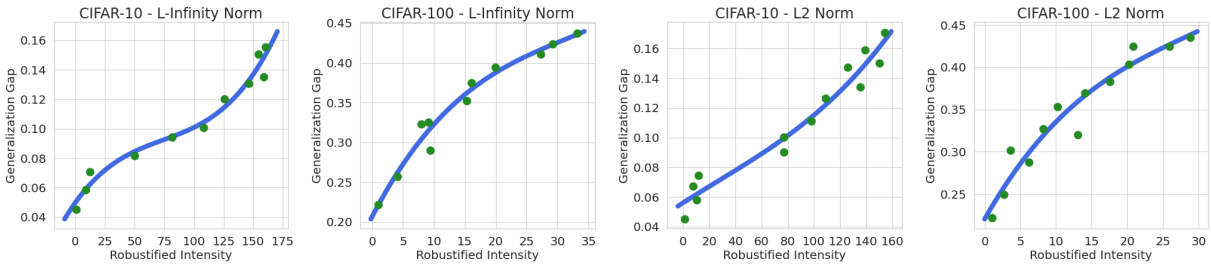


Figure 5: Plots of generalization gap vs. empirical robustified intensity. The datasets and PGD metric norms of the four plots are respectively (1) CIFAR-10 and L_∞ norm; (2) CIFAR-100 and L_∞ norm; (3) CIFAR-10 and L_2 norm; and (4) CIFAR-100 and L_2 norm. Fourth-order polynomial regression is used for curve fitting in each figure. Based on these figures, we find that generalization gap has a positive correlation with robustified intensity, which demonstrates a negative correlation between privacy preservation and adversarial robustness.

Proof of Theorem 6.2. Combining Lemma 6.1 and Lemma 6.3, we can directly prove Theorem 6.2. □

6.2 Tightness of generalization bounds

This section analyses the tightness of generalization bounds.

Dependency on the training sample size N. In Section 5, we have approximated the rate of adversarial training's differential privacy with respect to the training sample size N; see Remark 5.3. Combining Theorems 6.1 and 6.2, we can approximate the tightness of the high-probability generalization bound and the on-average generalization bound as the following two remarks.

Remark 6.2. *The high-probability generalization bound given by Theorem 6.1 is* $\mathcal{O}(1/\sqrt{N})$.

Remark 6.3. *The on-average generalization bound given by Theorem 6.2 is* $\mathcal{O}(\sqrt{\log N}/N)$.

Dependency on the model size. Our generalization bounds do not explicitly rely on the parameter size, which would be prohibitively large in deep learning. The only terms that could rely on the model size are the max gradient norms. We empirically investigated their magnitude. We trained Wide ResNets on CIFAR-10, CIFAR-100, under the frameworks of ERM and adversarial training with multiple different values of the radius ρ and two different metric norms in adversarial training, L_2 norm and L_∞ norm. The collected data is shown in fig. 4. The box plots clearly demonstrate that the gradient norms are very small comparing to the parameter size, which would be tens of millions.

Comparing with existing bounds. Existing works have proved generalization bounds based on hypothesis complexity. Yin et al. [94] prove an $\mathcal{O}(1/\sqrt{N})$ generalization bound for models learned by adversarial training, based on Rademacher complexity of the deep neural networks. Khim and Loh [37] also prove an $\mathcal{O}(1/\sqrt{N})$ generalization bound based on the Rademacher complexity of the hypothesis space. Tu et al. [84] prove an $\mathcal{O}(1/\sqrt{N})$ generalization bound based on the covering number of the hypothesis space. All these bounds heavily rely on the hypothesis complexity, which would, however, be prohibitively large in deep learning.

6.3 Experimental evidence

The empirical study on the generalization-robustness trade-off is based on Wide ResNet and datasets of CIFAR-10, CIFAR-100. The generalization abilities of all models are evaluated by the generalization

gap, which is defined to be the difference between the training accuracy and test accuracy. For more implementation details, please see Section 8.

We collect the generalization gaps and empirical robustified intensities of all models. Based on the collected data, four plots are shown in fig. 5. Fourth-order polynomial regression is employed for curve fitting. A larger generalization gap implies a worse generalizability. From the plots, one major observation is obtained: the generalization gap has a clear positive correlation with the robustified intensity, which verifies the generalization-robustness trade-off.

7 Proofs

In this section, we provide detail proofs omitted in the previous sections.

7.1 Proof of Theorem 4.2

We first recall additional preliminaries that necessary in the proofs.

Suppose every example z is independently and identically (i.i.d.) sampled from the data distribution is $\mathcal{D}$; *i.e.*, $z \sim \mathcal{D}$. Thus, the training sample set $S \sim \mathcal{D}^N$, where N is the training sample size.

Besides, we need the following two definitions in the rest of the paper.

Definition 7.1 (Ball and sphere)**.** *The ball in space $\mathcal{H}$ centered at point $x \in \mathcal{H}$ of radius $r > 0$ in term of norm $\|\cdot\|$ is denoted by*

$$\mathbb{B}_h(r) = \{x : \|x - h\| \leq r\}.$$

The sphere $\partial\mathbb{B}_h(r)$ of ball $\mathbb{B}_h(r)$ is defined as below,

$$\partial\mathbb{B}_h(r) = \{x : \|x - h\| = r\}.$$

Definition 7.2 (Complementary set)**.** *For a subset $A \subset \mathcal{H}$ of a space $\mathcal{H}$, its complementary set A^c is defined as below,*

$$A' = \{h : h \in \mathcal{H}, h \notin A\}.$$

Proof of Theorem 4.2. We only need to prove that almost surely

$$\lim_{\tau\to\infty} \max_{(x_i,y_i)\in\mathcal{B}} \left\| \nabla_\theta \max_{\|x_i'-x_i\|\leq\rho} l\left(h_\theta\left(x_i'\right), y_i\right) \right\| = \max_{x,y} \left\| \nabla_\theta \max_{\|x'-x\|\leq\rho} l\left(h_\theta\left(x'\right), y\right) \right\|, \tag{6}$$

and almost surely

$$\lim_{\tau\to\infty} \max_{(x_i,y_i)\in\mathcal{B}} \|\nabla_\theta l\left(h_\theta\left(x_i\right), y_i\right)\| = \max_{x,y} \|\nabla_\theta l\left(h_\theta\left(x\right), y\right)\|. \tag{7}$$

We first prove that for any positive real $\rho > 0$,

$$g(\theta, z) = \nabla_\theta \max_{x'\in\mathbb{B}_x(\rho)} l\left(h_\theta\left(x'\right), y\right) \tag{8}$$

is a continuous function with respect to $z = (x, y)$, where

$$\mathbb{B}_x(\rho) = \{x' : \|x - x'\| \leq \rho\}$$

is a ball centered at x with radius of ρ.

Fixing $y \in \mathcal{Y}$, define

$$T_y(x) = \arg \max_{x' \in \mathbb{B}_x(\rho)} \ell(h_\theta(x'), y)$$

as a mapping from $\mathcal{X}$ to $\mathcal{X}$. We will prove $T_y(x)$ is continuous with respect to (x, y) by reduction to absurdity. Suppose there exists a sequence

$$\{z_i = (x^i, y^i)\}_{i=1}^{\infty}, \ \lim_{i \to \infty} z_i = z_0,$$

and a constant $\varepsilon_A > 0$ such that

$$\|T_{y^i}(x^i) - T_{y^0}(x^0)\| \geq \varepsilon_A.$$

Since $\{T_{y^i}(x^i)\}_{i=1}^{\infty}$ is a bounded set, there exists an increasing subsequence $\{k_i\}_{i=1}^{\infty} \subseteq Z^+$ such that $\{T_{y_{k_i}}(x_{k_i})\}_{i=1}^{\infty}$ converges to some point T_∞. Then, we have that

$$T_\infty \in \cup_{i=1}^{\infty} \cap_{j=i}^{\infty} \mathbb{B}_{x_{k_i}}(\rho) \subset \mathbb{B}_{x^0}(\rho).$$

Furthermore, for any $\varepsilon \geq 0$, there exists a $\delta > 0$, such that for any $x \in \mathbb{B}_{T_{y^0}(x^0)}(\delta)$,

$$\ell(h_\theta(x), y^0)) \geq \ell(h_\theta(T_{y^0}(x^0)), y^0) - \varepsilon.$$

In case $T_{y^0}(x^0) \in \partial \mathbb{B}_{x^0}(\rho)$ such that $T_{y^0}(x^0) \not\subset \cap_{i=1}^{\infty} \mathbb{B}_{x_{k_i}}(\rho)$, let $x' \in \mathbb{B}_{T_{y^0}(x^0)}(\delta)$ be an inner point of $\mathbb{B}_{x^0}(\rho)$. When i is large enough, we have $x' \in \mathbb{B}_{x_{k_i}}(\rho)$, which yields

$$\ell(h_\theta(x'), y_{k_i}) \leq \ell(h_\theta(T_{y_{k_i}}(x_{k_i})), y_{k_i}).$$

Let i approaches ∞, we then have

$$\ell(h_\theta(T_{y^0}(x^0), y^0) - \varepsilon \leq \ell(h_\theta(x'), y^0) \leq \ell(h_\theta(T_\infty), y^0).$$

Since ε is arbitrarily selected, we then have

$$\ell(h_\theta(T_{y^0}(x^0), y^0)) \leq \ell(h_\theta(T_\infty), y^0) \leq \ell(h_\theta(T_{y^0}(x^0), y^0)).$$

Therefore, $T_\infty = T_{y^0}(x^0)$, which leads to a contradictary since

$$\|T_{y^i}(x^i) - T_{y^0}(x^0)\| \geq \varepsilon_A.$$

Since $g(\theta, z)$ can be rewritten as

$$g(\theta, z) = \nabla_\theta \max_{x' \in \mathbb{B}_x(\rho)} l\left(h_\theta\left(x'\right), y\right) = \nabla_\theta \ell(h_\theta(T_y(x)), y),$$

by Assumption 4.1, we have $g(\theta, z)$ is continuous with respect to z.

Now we can prove eq. (6) and eq. (7). As for eq. (6), there exist $z_0 = (x^0, y^0)$ such that

$$\left\| \nabla_\theta \max_{\|x' - x^0\| \leq \rho} l\left(h_\theta\left(x'\right), y^0\right) \right\| \quad = \max_{(x,y)} \left\| \nabla_\theta \max_{\|x' - x\| \leq \rho} l\left(h_\theta\left(x'\right), y\right) \right\|.$$

For any $\varepsilon > 0$, since $g(\theta, z)$ is continuous with respect to z, there exists a $\delta > 0$, such that for any $(x', y') \in \mathbb{B}_{(x^0,y^0)}(\delta)$,

$$g(\theta, (x', y')) \geq g(\theta, (x^0, y^0)) - \varepsilon.$$

Therefore,

$$\{(x, y) : g(\theta, (x, y)) < g(\theta, (x^0, y^0)) - \varepsilon\} \subset \left(\mathbb{B}_{(x^0,y^0)}(\delta)\right)^c,$$

and we have that

$$\begin{aligned}
&\mathbb{P}_{\mathcal{B}\sim\mathcal{D}^\tau}\left(\max_{z_i\in\mathcal{B}} g(\theta, z_i) < \max_z g(\theta, z) - \varepsilon\right) \\
\leq &\mathbb{P}_{\mathcal{B}\sim\mathcal{D}^\tau}\left(\max_{z_i\in\mathcal{B}} g(\theta, z_i) < g(\theta, z_0) - \varepsilon\right) \\
\leq &\mathbb{P}_{\mathcal{B}\sim\mathcal{D}^\tau}\left(\mathcal{B} \cap \mathbb{B}_{(x^0,y^0)}(\delta) = \emptyset\right) \\
= &\left(1 - \mathbb{P}_{z\sim\mathcal{D}}\left(z \in \mathbb{B}_{(x^0,y^0)}(\delta)\right)\right)^\tau.
\end{aligned}$$

As $\tau \to \infty$, we have

$$\lim_{\tau\to\infty} \mathbb{P}_{\mathcal{B}\sim\mathcal{D}^\tau}\left(\max_{z_i\in\mathcal{B}} g(\theta, z_i) \leq \max_z g(\theta, z) - \varepsilon\right) = 0.$$

Since ε is arbitrarily selected, we have

$$\lim_{\tau\to\infty} \mathbb{P}_{\mathcal{B}\sim\mathcal{D}^\tau}\left(\max_{z_i\in\mathcal{B}} g(\theta, z_i) \leq \max_z g(\theta, z)\right) = 0,$$

which proves eq. (6).

Replacing $g(\theta, z) = \nabla_\theta \ell(h_\theta(x), y)$, we can prove eq. (7) following the same routine.

The proof is completed. □

7.2 Proof of Theorem 5.1

This section proves Theorem 5.1. We first prove two lemmas.

Practically, high-probability approximations of ε-differential privacy are easier to be obtained from concentration inequalities. Lemma 7.1 presents a relationship from high-probability approximations of ε-differential privacy to approximations of (ε, δ)-differential privacy. Similar arguments are used in some related works; see, for example, the proof of Theorem 3.20 in [18]. Here, we give a detailed proof to make this paper completed.

Lemma 7.1. *Suppose $\mathcal{A} : \mathcal{Z}^N \to \mathcal{H}$ is a stochastic algorithm, whose output hypothesis learned on the training sample set S is $\mathcal{A}(S)$. For any hypothesis $h \in \mathcal{H}$, if for probability at least $1 - \delta$ over the randomness of $\mathcal{A}(S)$,*

$$\log\left[\frac{\mathbb{P}\left[\mathcal{A}(S) = h\right]}{\mathbb{P}\left[\mathcal{A}(S') = h\right]}\right] \leq \varepsilon, \tag{9}$$

the algorithm $\mathcal{A}$ is (ε, δ)-differentially private.

Proof. After rearranging eq. (9), we have that for probability at least $1 - \delta$,

$$\mathbb{P}\left[\mathcal{A}(S) = h\right] \leq \mathbb{P}\left[\mathcal{A}(S') = h\right] e^\varepsilon, \tag{10}$$

For the brevity, we define an event as follows,

$$B_0 = \left\{ h : \log \left[\frac{\mathbb{P}\left[\mathcal{A}(S) = h\right]}{\mathbb{P}\left[\mathcal{A}(S') = h\right]} \right] \leq \varepsilon \right\}.$$

Also, define that

$$B_0^c = \mathcal{H} - B_0.$$

Apparently, for any subset $B \in \mathcal{H}$,

$$\begin{aligned} \mathbb{P}\left[\mathcal{A}(S) \in B_0 \cap B\right] \leq & \mathbb{P}\left[\mathcal{A}(S') \in B_0 \cap B\right] e^{\varepsilon}, & (11) \\ \mathbb{P}\left[\mathcal{A}(S) \in B_0\right] \geq & 1 - \delta, \\ \mathbb{P}\left[\mathcal{A}(S) \in B_0^c\right] \leq & \delta. & (12) \end{aligned}$$

Then, for any subset $B \in \Theta$, we have that

$$\begin{aligned} & \mathbb{P}\left[\mathcal{A}(S) \in B\right] \\ = & \mathbb{P}\left[\mathcal{A}(S) \in B \cap (B_0 \cup B_0^c)\right] \\ = & \mathbb{P}\left[\mathcal{A}(S) \in B \cap B_0\right] + \mathbb{P}\left[\mathcal{A}(S) \in B \cap B_0^c\right] \\ \leq & \mathbb{P}\left[\mathcal{A}(S) \in B \cap B_0\right] + \mathbb{P}\left[\mathcal{A}(S) \in B_0^c\right]. \end{aligned}$$

Combining eqs. (10), (11), and (12), we have that

$$\begin{aligned} & \mathbb{P}\left[\mathcal{A}(S) \in B\right] \\ \leq & e^{\varepsilon} \mathbb{P}\left[\mathcal{A}(S') \in B \cap B_0\right] + \delta \\ \leq & e^{\varepsilon} \mathbb{P}\left[\mathcal{A}(S') \in B\right] + \delta. \end{aligned}$$

Therefore, the stochastic algorithm $\mathcal{A}$ is (ε, δ)-differentially private.

The proof is completed. □

We now prove the Theorem 5.1.

Proof of Theorem 5.1. We assume that the gradients calculated from random sampled mini batch $\mathcal{B}$ with size τ are random variables drawn from a Laplace distribution (see a justification in the main text):

$$\frac{1}{\tau} \sum_{z \in \mathcal{B}} \nabla_\theta \max_{\|x' - x\| \leq \rho} \ell(\theta, x, y) \sim \mathrm{Lap}\left(\nabla_\theta \hat{\mathcal{R}}_S^A(\theta), b \right).$$

Correspondingly, its counterpart on the training sample set S' is as follows,

$$\frac{1}{\tau} \sum_{z \in \mathcal{B}'} \nabla_\theta \max_{\|x' - x\| \leq \rho} \ell(\theta, x, y) \sim \mathrm{Lap}\left(\nabla_\theta \hat{\mathcal{R}}_{S'}^A(\theta), b \right),$$

where $\mathcal{B}'$ is uniformly sampled from S' with size τ.

The output hypothesis is uniquely indexed by the weight. Specifically, we denote the weight after the t-th iteration as θ_{t+1}^A. Furthermore, the weight updates $\Delta\theta_t^A = \theta_{t+1}^A - \theta_t^A$ are uniquely determined by the

gradients. Therefore, we can calculate the probability of the gradients to approximate the differential privacy. For any $\hat{g}_t^A$,

$$\begin{aligned}
&\log\left[\frac{p\left[\mathrm{Lap}(\nabla_\theta\hat{\mathcal{R}}_S^A(\theta_t^A),b)=\hat{g}_t^A\right]}{p\left[\mathrm{Lap}(\nabla_\theta\hat{\mathcal{R}}_{S'}^A(\theta_t^A),b)=\hat{g}_t^A\right]}\right]\\
=&\log\left[\frac{\exp\left\{-\left\|\nabla_\theta\hat{\mathcal{R}}_S^A(\theta_t^A)-\hat{g}_t^A\right\|/b\right\}}{\exp\left\{-\left\|\nabla_\theta\hat{\mathcal{R}}_{S'}^A(\theta_t^A)-\hat{g}_t^A\right\|/b\right\}}\right]\\
=&\frac{1}{b}\left[-\left\|\nabla_\theta\hat{\mathcal{R}}_S^A(\theta_t^A)-\hat{g}_t^A\right\|+\left\|\nabla_\theta\hat{\mathcal{R}}_{S'}^A(\theta_t^A)-\hat{g}_t^A\right\|\right].
\end{aligned}\tag{13}$$

Define that

$$L_t^A=\max_{(x,y)}\left\|\nabla_\theta\max_{\|x'-x\|\le\rho}\ell(\theta_t^A,x',y)\right\|,$$

and

$$v=\nabla_\theta\hat{\mathcal{R}}_{S'}^A(\theta_t^A)-\nabla_\theta\hat{\mathcal{R}}_S^A(\theta_t^A).$$

Because there only one pair of examples is different between the training sample set pair S and S', we have that,

$$\|v\|\le\frac{2L_A}{N}.\tag{14}$$

Combining eqs. (13) and (14), we have that

$$\begin{aligned}
&\log\left[\frac{p\left[\mathrm{Lap}(\nabla_\theta\hat{\mathcal{R}}_S^A(\theta_t^A),b)=\hat{g}_t^A\right]}{p\left[\mathrm{Lap}(\nabla_\theta\hat{\mathcal{R}}_{S'}^A(\theta_t^A),b)=\hat{g}_t^A\right]}\right]\\
=&\frac{1}{b}\left[-\left\|\nabla_\theta\hat{\mathcal{R}}_S^A(\theta_t^A)-\hat{g}_t^A\right\|+\left\|\nabla_\theta\hat{\mathcal{R}}_{S'}^A(\theta_t^A)-\hat{g}_t^A\right\|\right]\\
=&\frac{2L_A}{Nb}.
\end{aligned}\tag{15}$$

Since $L_t^A=I_tL_t^{\mathrm{ERM}}$, we have that

$$\log\left[\frac{p\left[\mathrm{Lap}(\nabla_\theta\hat{\mathcal{R}}_S^A(\theta_t^A),b)=\hat{g}_t^A\right]}{p\left[\mathrm{Lap}(\nabla_\theta\hat{\mathcal{R}}_{S'}^A(\theta_t^A),b)=\hat{g}_t^A\right]}\right]\le\frac{2L_t^{\mathrm{ERM}}}{Nb}I_t.$$

Define that

$$\varepsilon_t=\frac{2L_t^{\mathrm{ERM}}}{Nb}I_t.$$

Applying Lemma 5.1 with $\varepsilon_1,\cdots,\varepsilon_T$ and $\delta=\frac{\delta'}{N}$, the proof is completed. □

7.3 Proof of Theorem 5.2

Proof of Theorem 5.2. By Theorem 5.1, we have that the adversarial training is $(\varepsilon_A, \delta_A)$-differentially private, where

$$\varepsilon_A = \sqrt{2\log\frac{N}{\delta'}\sum_{t=1}^{T}\varepsilon_t^2} + \sum_{t=1}^{T}\varepsilon_t\frac{e^{\varepsilon_t}-1}{e^{\varepsilon_t}+1},$$

and $\delta_A = \frac{\delta'}{N}$, $\varepsilon_t = \frac{2L_t^{\mathrm{ERM}}}{Nb}I_t$.

To bridge the gap between differential privacy and the robustified intensity $I_{1:T}$, we then bound ε_A as follows,

$$\begin{aligned}\varepsilon_A &\le \sqrt{2\log\frac{N}{\delta'}\sum_{t=1}^{T}\varepsilon_t^2} + \frac{1}{2}\sum_{t=1}^{T}\varepsilon_t(e^{\varepsilon_t}-1)\\ &= \sqrt{2\log\frac{N}{\delta'}\sum_{t=1}^{T}\varepsilon_t^2} + \mathcal{O}\left(\frac{1}{N^2}\right). \end{aligned} \tag{16}$$

Notice that the first term in eq. (16) is $\mathcal{O}(\sqrt{\log N}/N)$, which suggests that eq. (16) is dominated by its first term and the factor $\sum_{t=1}^{T}\varepsilon_t^2$ is a good indicator to measure the privacy preserving ability of the adversarial training. Therefore, we further bound the factor $\sum_{t=1}^{T}\varepsilon_t^2$ as follows,

$$\begin{aligned}\sum_{t=1}^{T}\varepsilon_t^2 &= \sum_{t=1}^{T}\left(\frac{2L_t^{ERM}}{Nb}I_t\right)^2\\ &\le \sqrt{\sum_{t=1}^{T}\left(\frac{2L_t^{ERM}}{Nb}\right)^4}\cdot\sqrt{\sum_{t=1}^{T}I_t^4} \qquad (17)\\ &= \sqrt{\left(\frac{2}{Nb}\right)^4\cdot T\cdot\left(L_{1:T}^{\mathrm{ERM}}\right)^4}\cdot\sqrt{T\cdot I_{1:T}^4}\\ &= T\cdot\left(\frac{2L_{1:T}^{\mathrm{ERM}}}{Nb}I_{1:T}\right)^2 = T\cdot\varepsilon_{1:T}^2, \qquad (18)\end{aligned}$$

where eq. (17) follows by CauchySchwarz inequality. Inserting eq. (18) into eq. (16), we have that

$$\varepsilon_A \le \varepsilon_{1:T}\sqrt{2T\log\frac{N}{\delta'}} + \mathrm{O}\left(\frac{1}{N^2}\right).$$

Finally, defining $\varepsilon := \varepsilon_{1:T}\sqrt{2T\log\frac{N}{\delta'}} + \mathcal{O}\left(\frac{1}{N^2}\right)$ and $\delta := \delta_A = \frac{\delta'}{N}$, then according to the definition of differential privacy, we can conclude that the adversarial training is also (ε, δ)-differentially private, which completes the proof. □

7.4 Proof of Lemma 6.1

This section proves the relationship between differential privacy and uniform stability.

We first prove a weaker version of Lemma 6.1 when algorithm $\mathcal{A}$ has ε-pure differential privacy.

Lemma 7.2. *Suppose a machine learning algorithm $\mathcal{A}$ is ε-differentially private. Assume the loss function l is upper bounded by a positive real constant $M > 0$. Then, the algorithm $\mathcal{A}$ is uniformly stable,*

$$\left|\mathbb{E}_{\mathcal{A}(S)}\ell(\mathcal{A}(S), Z) - \mathbb{E}_{\mathcal{A}(S')}\ell(\mathcal{A}(S'), Z)\right| \leq M(1 - e^{-\varepsilon}).$$

Proof. Let set B defined as $B = \{h \in H : \ell(h, z) > t\}$, where t is an arbitrary real. Then, for any $t \in \mathbb{R}$,

$$\mathbb{P}_{\mathcal{A}(S)}(\mathcal{A}(S) \in B) \leq e^{\varepsilon}\mathbb{P}_{\mathcal{A}(S')}(\mathcal{A}(S') \in B). \tag{19}$$

By rearranging eq. (19), we have

$$e^{-\varepsilon}\mathbb{P}_{\mathcal{A}(S)}(\mathcal{A}(S) \in B) \leq \mathbb{P}_{\mathcal{A}(S')}(\mathcal{A}(S') \in B),$$

and

$$(e^{-\varepsilon} - 1)\mathbb{P}_{\mathcal{A}(S)}(\mathcal{A}(S) \in B) \leq \mathbb{P}_{\mathcal{A}(S')}(\mathcal{A}(S') \in B) - \mathbb{P}_{\mathcal{A}(S)}(\mathcal{A}(S) \in B).$$

Since $\varepsilon > 0$, we have $e^{-\varepsilon} < 1$. Therefore,

$$(e^{-\varepsilon} - 1) \leq \mathbb{P}_{\mathcal{A}(S')}(\mathcal{A}(S') \in B) - \mathbb{P}_{\mathcal{A}(S)}(\mathcal{A}(S) \in B). \tag{20}$$

Eq. (20) stands for every neighbor sample set pair S and S'. Thus,

$$e^{-\varepsilon} - 1 \leq \min_{S \text{ and } S' \text{ neighbor}}(\mathbb{P}_{\mathcal{A}(S')}(\mathcal{A}(S') \in B) - \mathbb{P}_{\mathcal{A}(S)}(\mathcal{A}(S) \in B)).$$

Therefore,

$$\max_{S \text{ and } S' \text{ neighbor}} \left|\left[\mathbb{P}_{\mathcal{A}(S')}(\mathcal{A}(S') \in B) - \mathbb{P}_{\mathcal{A}(S)}(\mathcal{A}(S) \in B)\right]\right| \leq 1 - e^{-\varepsilon}.$$

Thus,

$$\begin{aligned}
&|\mathbb{E}_{\mathcal{A}(S')}\ell(\mathcal{A}(S'), z) - \mathbb{E}_{\mathcal{A}(S)}\ell(\mathcal{A}(S), z)| \\
=&\left|\int \ell(\mathcal{A}(S), z)\, d\mathbb{P}_{\mathcal{A}(S)} - \int \ell(\mathcal{A}(S'), z)\mathrm{d}\, \mathbb{P}_{\mathcal{A}(S')}\right| \\
\overset{(*)}{\leq}& \max\{I_1, I_2\} \\
\leq& M(1 - e^{-\varepsilon}),
\end{aligned}$$

where I_1 and I_2 in inequality $(*)$ is defined as

$$\begin{aligned}
I_1 &= \int_{\mathbb{P}_{\mathcal{A}(S)} > \mathbb{P}_{\mathcal{A}(S')}} \ell(\mathcal{A}(S), z)\left(\mathrm{d}\mathbb{P}_{\mathcal{A}(S)} - \mathrm{d}\mathbb{P}_{\mathcal{A}(S')}\right), \\
I_2 &= \int_{\mathbb{P}_{\mathcal{A}(S)} \leq \mathbb{P}_{\mathcal{A}(S')}} \ell(\mathcal{A}(S), z)\left(\mathrm{d}\mathbb{P}_{\mathcal{A}(S')} - \mathrm{d}\mathbb{P}_{\mathcal{A}(S)}\right).
\end{aligned}$$

The proof is completed. □

Then, we prove Lemma 6.1 using a different method.

Proof of Lemma 6.1. As the algorithm $\mathcal{A}$ is (ε, δ)-differentially private, we have

$$\mathbb{P}_{\mathcal{A}(S)}(\mathcal{A}(S) \in B) \leq e^{\varepsilon}\mathbb{P}_{\mathcal{A}(S')}(\mathcal{A}(S') \in B) + \delta,$$

where the subset B is arbitrary from the hypothesis space $\mathcal{H}$. Let $B = \{h \in H : \ell(h, z) > t\}$. Then we have the following inequality,

$$\mathbb{P}_{\mathcal{A}(S)}(\ell(\mathcal{A}(S), z) > t) \leq e^{\varepsilon}\mathbb{P}_{\mathcal{A}(S')}(\ell(\mathcal{A}(S'), z) > t) + \delta. \tag{21}$$

Additionally, $\mathbb{E}_{\mathcal{A}(S)}\ell(\mathcal{A}(S), z)$ is calculated as follows,

$$\mathbb{E}_{\mathcal{A}(S)}\ell(\mathcal{A}(S), z) = \int_0^M \mathbb{P}_{\mathcal{A}(S)}(\ell(\mathcal{A}(S), z) > t)dt.$$

Applying eq. (21), we have

$$\begin{aligned}
&\mathbb{E}_{\mathcal{A}(S)}\ell(\mathcal{A}(S), z) \\
=&\int_0^M \mathbb{P}_{\mathcal{A}(S)}(\ell(\mathcal{A}(S), z) > t)dt \\
\leq& e^{\varepsilon}\int_0^M \mathbb{P}_{\mathcal{A}(S')}(\ell(\mathcal{A}(S'), z) > t)dt + M\delta \\
=& e^{\varepsilon}\mathbb{E}_{\mathcal{A}(S')}\ell(\mathcal{A}(S'), z) + M\delta.
\end{aligned} \tag{22}$$

Rearranging eq. (22), we have

$$e^{-\varepsilon}\mathbb{E}_{\mathcal{A}(S)}\ell(\mathcal{A}(S), z) \leq \mathbb{E}_{\mathcal{A}(S')}\ell(\mathcal{A}(S'), z) + e^{-\varepsilon}M\delta,$$

and

$$(e^{-\varepsilon} - 1)\mathbb{E}_{\mathcal{A}(S)}\ell(\mathcal{A}(S), z) \leq \mathbb{E}_{\mathcal{A}(S')}\ell(\mathcal{A}(S'), z) - \mathbb{E}_{\mathcal{A}(S)}\ell(\mathcal{A}(S), z) + e^{-\varepsilon}M\delta.$$

Therefore,

$$\mathbb{E}_{\mathcal{A}(S')}\ell(\mathcal{A}(S'), z) - \mathbb{E}_{\mathcal{A}(S)}\ell(\mathcal{A}(S), z) \leq e^{-\varepsilon}M\delta + (1 - e^{-\varepsilon})\mathbb{E}_{\mathcal{A}(S)}\ell(\mathcal{A}(S), z).$$

Similarly, we can get the following inequality,

$$-\mathbb{E}_{\mathcal{A}(S')}\ell(\mathcal{A}(S'), z) + \mathbb{E}_{\mathcal{A}(S)}\ell(\mathcal{A}(S), z) \leq e^{-\varepsilon}M\delta + (1 - e^{-\varepsilon})\mathbb{E}_{\mathcal{A}(S)}\ell(\mathcal{A}(S), z).$$

Thus,

$$\left|\mathbb{E}_{\mathcal{A}(S)}\ell(\mathcal{A}(S), Z) - \mathbb{E}_{\mathcal{A}(S')}\ell(\mathcal{A}(S'), Z)\right| \leq M\delta e^{-\varepsilon} + M(1 - e^{-\varepsilon}).$$

The proof is completed. □

8 Experimental implementation details

This section presents implementation details of our experiments, including the settings of adversarial training, the calculation of empirical composite robustified intensity and gradient noise, and the implementation of membership inference attack. The running time of both ERM and adversarial training is also presented. The training code, learned models, and collected data are available at `https://github.com/fshp971/RPG`.

8.1 Datasets

We use the CIFAR-10 and CIFAR-100 datasets [41] in our experiments. Both datasets contain $60,000$ 32×32 color images, where $50,000$ are training images and $10,000$ are test images. The images in CIFAR-10 are grouped into 10 different categories, while images in CIFAR-100 have 100 different categories. No pre-processing is involved. The datasets can be downloaded from `https://www.cs.toronto.edu/~kriz/cifar.html`.

8.2 Neural network architectures

We use a 34-layer Wide ResNet [95], WRN-34-10, in all experiments. The detailed architectures of WRN-34-10 are presented in Table 1, where "conv x-c" represents a convolutional layer with kernel size $x \times x$ and c output channels, "fc-c" represents a fully-connected layer with c output channels, and "$[\cdot]$" represents the residual block named basic block [33]. Each convolution layer is followed by batch normalization and then ReLU activation.

Table 1: Neural network architecture of WRN-34-10.

WRN-34-10
conv3-16
$\begin{bmatrix} \text{conv3-160} \\ \text{conv3-160} \end{bmatrix} \times 5$
$\begin{bmatrix} \text{conv3-320} \\ \text{conv3-320} \end{bmatrix} \times 5$
$\begin{bmatrix} \text{conv3-640} \\ \text{conv3-640} \end{bmatrix} \times 5$
avgpool
fc-10 or fc-100

8.3 Projected gradient descent

Projected gradient descent (PGD) [53] performs K iterative gradient ascent to search an example that maximizes the training loss. When using the L_∞ norm, the k-th update in PGD is as follows,

$$x_k = \prod_{\|x'-x\|_\infty \le \rho} \left[x_{k-1} + \alpha \cdot \text{sign}\left(\nabla_x l\left(h_\theta(x_{k-1}), y \right) \right) \right],$$

where x_k is the adversarial example obtained in the k-th iteration, α is the step size, and $\prod_{x':\|x'-x\|_\infty\leq\rho}$ means that the projection is calculated in the ball sphere $\mathbb{B}(x,\rho)=\{x':\|x'-x\|_\infty\leq\rho\}$. Besides, when using L_2 norm as the metric, the k-th update is as follows,

$$x_k=\prod_{x':\|x'-x\|_2\leq\rho}\left[x_{k-1}+\alpha\cdot\nabla_x l\left(h_\theta(x_{k-1}),y\right)\right].$$

The iterations number K is set as 8. The step size α is set as $\rho/4$.

8.4 Training settings

All the models are trained by SGD for $80,000$ iterations. The momentum factor is set as 0.9, the weight decay factor is set as 0.0002, and the batch size is set as 128. The learning rate is initialized as 0.1 and then decays by 0.1 every $30,000$ iterations. The list of the radius ρ is presented in Table 2 as well as other factors.

Table 2: Details of different experimental settings.

Setting	Dataset	Norm of PGD	Radius ρ
A	CIFAR-10	L_∞	$\{\frac{i}{255}:i=0,\ 1,\ \ldots,\ 10\}$
B	CIFAR-100	L_∞	$\{\frac{i}{255}:i=0,\ 1,\ \ldots,\ 10\}$
C	CIFAR-10	L_2	$\{\frac{25\cdot i}{255}:i=0,\ 1,\ \cdots,\ 12\}$
D	CIFAR-100	L_2	$\{\frac{25\cdot i}{255}:i=0,\ 1,\ \cdots,\ 12\}$

8.5 Calculation of empirical robustified intensity

Calculating the empirical robustified intensity would be of a considerable high computational cost. We thus apply a sparse version in our experiments. We calculate the empirical single-iteration robustified intensities every m iterations. Thus, we obtain $\lfloor T/m\rfloor$ empirical single-iteration robustified intensities $\hat{I}_m,\hat{I}_{2m},\cdots,\hat{I}_{\lfloor T/m\rfloor\cdot m}$. The sparse version of empirical robustified intensity is as below,

$$\left(\frac{1}{\lfloor T/m\rfloor}\sum_{i=1}^{\lfloor T/m\rfloor}\hat{I}_{m\cdot i}^4\right)^{\frac{1}{4}}.$$

In our experiments, the calculation interval m is set as 100.

8.6 Calculation of gradient noise

We apply a five-step approach to estimate the gradient noise for a specific parameter θ: (1) calculate the gradient $\nabla_\theta\hat{\mathcal{R}}_S(\theta)$ of the full training set S; (2) randomly draw a mini batch from the training set and then calculate the gradient whereon; (3) calculate the difference between the mini-batch gradient and its full-batch counterpart; (4) randomly sample a set of components of the difference vector in the previous step; and (5) normalize all components in the sampled vector by their standard deviation.

8.7 Membership inference attack

We employ a threshold-based version of membership inference attack [93] to empirically assess the privacy-preserving abilities. Given a training set S_{train}, a test set S_{test}, and a trained model $h_\theta(\cdot)$. Suppose a data point (x, y) comes from S_{train} or S_{test} with equal probabilities. Then, the membership inference attack accuracy with a threshold ζ is calculated as follows,

$$Acc(\zeta) = \frac{1}{2} \times \left(\frac{\sum_{(x,y)\in S_{\text{train}}} \mathbf{1}[h_\theta(x)_y \geq \zeta]}{|S_{\text{train}}|} + \frac{\sum_{(x,y)\in S_{\text{test}}} \mathbf{1}[h_\theta(x)_y < \zeta]}{|S_{\text{test}}|} \right),$$

where $h_\theta(x)_y$ is the output confidence for label y and $\mathbf{1}[\cdot]$ is the indicator function. Therefore, the goal of the threshold-based attack model is to find an optimal threshold ζ_{optim} that maximizes the attack accuracy, *i.e.*,

$$\zeta_{\text{optim}} = \arg\max_{\zeta} Acc(\zeta),$$

and this can be done by enumerating all possible threshold values ζ.

8.8 Hardware environment

All our experiments are conducted on a computing cluster with GPUs of NVIDIA® Tesla™ V100 16GB and CPUs of Intel® Xeon® Gold 6140 CPU @ 2.30GHz.

8.9 Running time

We estimate the experiment running time based on the log files, as shown in Table 3. The running time includes both the training time and the time of calculating the robustified intensities in every setting. It is worth noting that precise running time may vary depending on the specific experimental conditions such as the temperature of GPUs and the working load of computing cluster. Therefore, the estimated running time given here would be not precise.

Table 3: Estimated running time of each experimental setting. "AT" is for adversarial training.

	Setting A	Setting B	Setting C	Setting D
ERM	7.5 hrs	7.4 hrs	7.5 hrs	7.4 hrs
AT	53.1 hrs	49.8 hrs	56.4 hrs	52.4 hrs

9 Conclusion

This paper studies the privacy-preserving and generalization abilities of adversarial training. We prove that the adversarial robustness, privacy preservation, and generalization are interrelated from both theoretical and empirical perspectives. We define *robustified intensity* and design its empirical version, *empirical robustified intensity*, which is proved to be asymptotically consistent with the robustified intensity. We then prove that adversarial training scheme is (ε, δ)-differentially private, in which the magnitude of the differential privacy (ε, δ) has a positive correlation with the robustified intensity. Based on the privacy-robustness relationship, an $\mathcal{O}(\sqrt{\log N}/N)$ on-average generalization bound and a $\mathcal{O}(1/\sqrt{N})$ high-probability one for adversarial training are delivered (N is the training sample size). Extensive systematic experiments are conducted based

on network architecture Wide ResNet and datasets CIFAR-10, CIFAR-100. The results fully support our theories.

References

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *ACM SIGSAC Conference on Computer and Communications Security*, pages 308–318, 2016.

[2] Pathum Chamikara Mahawaga Arachchige, Peter Bertok, Ibrahim Khalil, Dongxi Liu, Seyit Camtepe, and Mohammed Atiquzzaman. Local differential privacy for deep learning. *IEEE Internet of Things Journal*, 2019.

[3] Shumeet Baluja and Ian Fischer. Learning to attack: Adversarial transformation networks. In *AAAI Conference on Artificial Intelligence*, volume 1, page 3, 2018.

[4] Peter L Bartlett, Dylan J Foster, and Matus J Telgarsky. Spectrally-normalized margin bounds for neural networks. In *Advances in Neural Information Processing Systems*, pages 6240–6249, 2017.

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

[6] Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Šrndić, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. Evasion attacks against machine learning at test time. In *European Conference on Machine Learning*, 2013.

[7] Anselm Blumer, Andrzej Ehrenfeucht, David Haussler, and Manfred K Warmuth. Learnability and the Vapnik-Chervonenkis dimension. *Journal of the ACM*, 36(4):929–965, 1989.

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

[9] Mark Bun and Thomas Steinke. Concentrated differential privacy: Simplifications, extensions, and lower bounds. In *Theory of Cryptography Conference*, pages 635–658, 2016.

[10] Kamalika Chaudhuri, Jacob Imola, and Ashwin Machanavajjhala. Capacity bounded differential privacy. *arXiv preprint arXiv:1907.02159*, 2019.

[11] Hongming Chen, Ola Engkvist, Yinhai Wang, Marcus Olivecrona, and Thomas Blaschke. The rise of deep learning in drug discovery. *Drug Discovery Today*, 23(6):1241–1250, 2018.

[12] Sizhe Chen, Zhengbao He, Chengjin Sun, Jie Yang, and Xiaolin Huang. Universal adversarial attack on attention and the resulting dataset damagenet. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[13] Paul Cuff and Lanqing Yu. Differential privacy as a mutual information constraint. In *ACM SIGSAC Conference on Computer and Communications Security*, pages 43–54, 2016.

[14] Hanjun Dai, Hui Li, Tian Tian, Xin Huang, Lin Wang, Jun Zhu, and Le Song. Adversarial attack on graph structured data. *arXiv preprint arXiv:1806.02371*, 2018.

[15] Richard M Dudley. The sizes of compact subsets of hilbert space and continuity of Gaussian processes. *Journal of Functional Analysis*, 1(3):290–330, 1967.

[16] Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toni Pitassi, Omer Reingold, and Aaron Roth. Generalization in adaptive data analysis and holdout reuse. In *Advances in Neural Information Processing Systems*, pages 2350–2358, 2015.

[17] Cynthia Dwork and Deirdre K Mulligan. It's not privacy, and it's not fair. *Stanford Law Review Online*, 66:35, 2013.

[18] Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014.

[19] Cynthia Dwork and Guy N Rothblum. Concentrated differential privacy. *arXiv preprint arXiv:1603.01887*, 2016.

[20] Weinan E, Chao Ma, Stephan Wojtowytsch, and Lei Wu. Towards a mathematical understanding of neural network-based machine learning: What we know and what we don't. *arXiv preprint arXiv:2009.10713*, 2020.

[21] Vitaly Feldman and Jan Vondrak. High probability generalization bounds for uniformly stable algorithms with nearly optimal rate. *arXiv preprint arXiv:1902.10710*, 2019.

[22] Thomas Fischer and Christopher Krauss. Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research*, 270(2):654–669, 2018.

[23] Joseph Geumlek, Shuang Song, and Kamalika Chaudhuri. Renyi differential privacy mechanisms for posterior sampling. In *Advances in Neural Information Processing Systems*, pages 5289–5298, 2017.

[24] Justin Gilmer, Ryan P Adams, Ian Goodfellow, David Andersen, and George E Dahl. Motivating the rules of the game for adversarial example research. *arXiv preprint arXiv:1807.06732*, 2018.

[25] Noah Golowich, Alexander Rakhlin, and Ohad Shamir. Size-independent sample complexity of neural networks. In *Annual Conference on Learning Theory*, pages 297–299, 2018.

[26] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[27] Moritz Hardt, Ben Recht, and Yoram Singer. Train faster, generalize better: Stability of stochastic gradient descent. In *International Conference on Machine learning*, pages 1225–1234, 2016.

[28] Nick Harvey, Christopher Liaw, and Abbas Mehrabian. Nearly-tight VC-dimension bounds for piecewise linear neural networks. In *Annual Conference on Learning Theory*, pages 1064–1068, 2017.

[29] David Haussler. Sphere packing numbers for subsets of the boolean n-cube with bounded Vapnik-Chervonenkis dimension. *Journal of Combinatorial Theory, Series A*, 69(2):217–232, 1995.

[30] Fengxiang He, Tongliang Liu, and Dacheng Tao. Control batch size and learning rate to generalize well: Theoretical and empirical evidence. In *Advances in Neural Information Processing Systems*, 2019.

[31] Fengxiang He and Dacheng Tao. Recent advances in deep learning theory. *arXiv preprint arXiv:2012.10931*, 2020.

[32] Fengxiang He, Bohan Wang, and Dacheng Tao. Tighter generalization bounds for iterative differentially private learning algorithms. *arXiv preprint arXiv:2007.09371*, 2020.

[33] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[34] Stanisław Jastrzebski, Zachary Kenton, Devansh Arpit, Nicolas Ballas, Asja Fischer, Yoshua Bengio, and Amos Storkey. Three factors influencing minima in sgd. *arXiv preprint arXiv:1711.04623*, 2017.

[35] Fazle Karim, Somshubra Majumdar, and Houshang Darabi. Adversarial attacks on time series. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[36] Michael Kearns and Dana Ron. Algorithmic stability and sanity-check bounds for leave-one-out cross-validation. *Neural Computation*, 11(6):1427–1453, 1999.

[37] Justin Khim and Po-Ling Loh. Adversarial risk bounds via function transformation. *arXiv preprint arXiv:1810.09519*, 2018.

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

[39] Vladimir Koltchinskii. Rademacher penalties and structural risk minimization. *IEEE Transactions on Information Theory*, 47(5):1902–1914, 2001.

[40] Vladimir Koltchinskii and Dmitriy Panchenko. Rademacher processes and bounding the risk of function learning. In *High Dimensional Probability II*, pages 443–457. Springer, 2000.

[41] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. 2009.

[42] Casimir A Kulikowski. Artificial intelligence methods and systems for medical consultation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (5):464–476, 1980.

[43] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*, 2016.

[44] Harold Kushner and G George Yin. *Stochastic approximation and recursive algorithms and applications*, volume 35. Springer Science & Business Media, 2003.

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

[46] Mathias Lecuyer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, and Suman Jana. On the connection between differential privacy and adversarial robustness in machine learning. *arXiv preprint arXiv:1802.03471*, 2018.

[47] Mathias Lecuyer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, and Suman Jana. Certified robustness to adversarial examples with differential privacy. In *IEEE Symposium on Security and Privacy*, pages 656–672, 2019.

[48] Bai Li, Changyou Chen, Wenlin Wang, and Lawrence Carin. Second-order adversarial attack and certifiable robustness. 2018.

[49] Tengyuan Liang, Tomaso Poggio, Alexander Rakhlin, and James Stokes. Fisher-rao metric, geometry, and complexity of neural networks. In *International Conference on Artificial Intelligence and Statistics*, pages 888–896, 2019.

[50] Jiachun Liao, Lalitha Sankar, Vincent YF Tan, and Flavio du Pin Calmon. Hypothesis testing under mutual information privacy constraints in the high privacy regime. *IEEE Transactions on Information Forensics and Security*, 13(4):1058–1071, 2017.

[51] Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen Awm Van Der Laak, Bram Van Ginneken, and Clara I Sánchez. A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42:60–88, 2017.

[52] Lennart Ljung, Georg Pflug, and Harro Walk. *Stochastic approximation and optimization of random systems*, volume 17. Birkhäuser, 2012.

[53] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

[54] Stephan Mandt, Matthew D Hoffman, and David M Blei. Stochastic gradient descent as approximate Bayesian inference. *Journal of Machine Learning Research*, 18(1):4873–4907, 2017.

[55] David A McAllester. PAC-Bayesian model averaging. In *Annual Conference of Learning Theory*, volume 99, pages 164–170, 1999.

[56] David A McAllester. Some PAC-Bayesian theorems. *Machine Learning*, 37(3):355–363, 1999.

[57] Ilya Mironov. Rényi differential privacy. In *IEEE Computer Security Foundations Symposium*, pages 263–275, 2017.

[58] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning*. MIT press, 2018.

[59] Wenlong Mou, Liwei Wang, Xiyu Zhai, and Kai Zheng. Generalization bounds of sgld for non-convex learning: Two theoretical viewpoints. In *Annual Conference On Learning Theory*, pages 605–638, 2018.

[60] Preetum Nakkiran. Adversarial robustness may be at odds with simplicity. *arXiv preprint arXiv:1901.00532*, 2019.

[61] Yurii E Nesterov. A method for solving the convex programming problem with convergence rate o (1/kˆ 2). In *Dokl. Akad. Nauk Sssr*, volume 269, pages 543–547, 1983.

[62] Behnam Neyshabur, Srinadh Bhojanapalli, and Nathan Srebro. A PAC-Bayesian approach to spectrally-normalized margin bounds for neural networks. *arXiv preprint arXiv:1707.09564*, 2017.

[63] Anh Nguyen, Jason Yosinski, and Jeff Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 427–436, 2015.

[64] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In *ACM on Asia Conference on Computer and Communications Security*, pages 506–519, 2017.

[65] Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z Berkay Celik, and Ananthram Swami. The limitations of deep learning in adversarial settings. In *IEEE European Symposium on Security and Privacy*, pages 372–387, 2016.

[66] NhatHai Phan, Ruoming Jin, My T Thai, Han Hu, and Dejing Dou. Preserving differential privacy in adversarial learning with provable robustness. *arXiv preprint arXiv:1903.09822*, 2019.

[67] Rafael Pinot, Florian Yger, Cédric Gouy-Pailler, and Jamal Atif. A unified view on differential privacy and robustness to adversarial examples. *arXiv preprint arXiv:1906.07982*, 2019.

[68] Tomaso Poggio, Andrzej Banburski, and Qianli Liao. Theoretical issues in deep networks. *Proceedings of the National Academy of Sciences*, 2020.

[69] Herbert Robbins and Sutton Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, pages 400–407, 1951.

[70] William H Rogers and Terry J Wagner. A finite sample distribution-free performance bound for local discrimination rules. *The Annals of Statistics*, pages 506–514, 1978.

[71] Ludwig Schmidt, Shibani Santurkar, Dimitris Tsipras, Kunal Talwar, and Aleksander Madry. Adversarially robust generalization requires more data. *Advances in Neural Information Processing Systems*, 31:5014–5026, 2018.

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

[73] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *IEEE Symposium on Security and Privacy (SP)*, pages 3–18, 2017.

[74] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, and Marc Lanctot. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484, 2016.

[75] Samuel L Smith and Quoc V Le. A Bayesian perspective on generalization and stochastic gradient descent. In *International Conference on Learning Representations*, 2018.

[76] Robert Snelick, Umut Uludag, Alan Mink, Mike Indovina, and Anil Jain. Large-scale evaluation of multimodal biometric authentication using state-of-the-art systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(3):450–455, 2005.

[77] Liwei Song, Reza Shokri, and Prateek Mittal. Privacy risks of securing machine learning models against adversarial examples. In *ACM SIGSAC Conference on Computer and Communications Security*, pages 241–257, 2019.

[78] Ke Sun, Zhanxing Zhu, and Zhouchen Lin. Towards understanding adversarial examples systematically: Exploring data size, task and model factors. *arXiv preprint arXiv:1902.11019*, 2019.

[79] Zehang Sun, George Bebis, and Ronald Miller. On-road vehicle detection: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(5):694–711, 2006.

[80] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

[81] Paul Tseng. An incremental gradient (-projection) method with momentum term and adaptive stepsize rule. *SIAM Journal on Optimization*, 8(2):506–531, 1998.

[82] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. In *International Conference on Learning Representations*, 2019.

[83] Zhuozhuo Tu, Fengxiang He, and Dacheng Tao. Understanding generalization in recurrent neural networks. In *International Conference on Learning Representations*, 2020.

[84] Zhuozhuo Tu, Jingwei Zhang, and Dacheng Tao. Theoretical analysis of adversarial learning: A minimax approach. In *Advances in Neural Information Processing Systems*, pages 12280–12290, 2019.

[85] Vladimir Vapnik. *Estimation of Dependences based on Empirical Data*. Springer Science & Business Media, 2006.

[86] Vladimir Vapnik. *The Nature of Statistical Learning Theory*. Springer Science & Business Media, 2013.

[87] Saurabh Verma and Zhi-Li Zhang. Stability and generalization of graph convolutional neural networks. In *ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1539–1548, 2019.

[88] Fei Wang, Noah Lee, Jianying Hu, Jimeng Sun, Shahram Ebadollahi, and Andrew F Laine. A framework for mining signatures from event sequences and its applications in healthcare data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(2):272–285, 2012.

[89] Hongjun Wang, Guanbin Li, Xiaobai Liu, and Liang Lin. A hamiltonian monte carlo method for probabilistic adversarial attack and learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[90] Weina Wang, Lei Ying, and Junshan Zhang. On the relation between identifiability, differential privacy, and mutual-information privacy. *IEEE Transactions on Information Theory*, 62(9):5018–5029, 2016.

[91] Yuxin Wen, Jiehong Lin, Ke Chen, CL Philip Chen, and Kui Jia. Geometry-aware generation of adversarial point clouds. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[92] Huan Xu, Constantine Caramanis, and Shie Mannor. Sparse algorithms are not stable: A no-free-lunch theorem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(1):187–193, 2011.

[93] Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. Privacy risk in machine learning: Analyzing the connection to overfitting. In *IEEE Computer Security Foundations Symposium*, pages 268–282, 2018.

[94] Dong Yin, Ramchandran Kannan, and Peter Bartlett. Rademacher complexity for adversarially robust generalization. In *International Conference on Machine learning*, pages 7085–7094, 2019.

[95] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

[96] Tianhang Zheng, Changyou Chen, and Kui Ren. Distributionally adversarial attack. In *AAAI Conference on Artificial Intelligence*, volume 33, pages 2253–2260, 2019.