Robustness and Accuracy Could Be Reconcilable by (Proper) Definition

Tianyu Pang [1] [2] Min Lin [2] Xiao Yang [1] Jun Zhu [1] Shuicheng Yan [2]

Abstract

The trade-off between robustness and accuracy has been widely studied in the adversarial literature. Although still controversial, the prevailing view is that this trade-off is inherent, either empirically or theoretically. Thus, we dig for the origin of this trade-off in adversarial training and find that it may stem from the improperly defined robust error, which imposes an inductive bias of local invariance — an overcorrection towards smoothness. Given this, we advocate employing local equivariance to describe the ideal behavior of a robust model, leading to a self-consistent robust error named SCORE. By definition, SCORE facilitates the reconciliation between robustness and accuracy, while still handling the worst-case uncertainty via robust optimization. By simply substituting KL divergence with variants of distance metrics, SCORE can be efficiently minimized. Empirically, our models achieve top-rank performance on RobustBench under AutoAttack. Besides, SCORE provides instructive insights for explaining the overfitting phenomenon and semantic input gradients observed on robust models.

1. Introduction

The trade-off between *adversarial* robustness and *clean* accuracy has been widely observed (Schmidt et al., 2018; Su et al., 2018; Zhang et al., 2019; Wang et al., 2020a;b). On some simple cases, this trade-off is even shown to provably exist (Tsipras et al., 2019; Nakkiran, 2019; Raghunathan et al., 2020; Javanmard et al., 2020; Yu et al., 2021). With that, many strategies have been proposed to alleviate this trade-off via, e.g., early-stopping (Rice et al., 2020; Zhang et al., 2020), instance reweighting (Balaji et al., 2019; Zhang et al., 2021a), and exploiting extra data (Alayrac et al., 2019; Carmon et al., 2019; Hendrycks et al., 2019), to name a few. Nevertheless, other findings show the opposite. Stutz et al. (2019) and Yang et al. (2020) argue that robustness and accuracy can both be achievable through manifold analyses or locally Lipschitz functions. Rozsa et al. (2016) and Gilmer et al. (2018) also reveal that better generalization helps robustness on both toy and large-scale datasets.

Given arguments on both sides, it is much debated whether the robustness-accuracy trade-off is intrinsically there. But before considering its existence, let us first reach a consensus on how a robust model is supposed to behave — the definition of robustness. A most popular one is made by Madry et al. (2018), who define robustness in terms of robust error formulated as a locally maximized loss function. This definition is widely adopted in adversarial training (AT) (Shafahi et al., 2019; Wong et al., 2020). Besides, there are also some other definitions. For example, Szegedy et al. (2014) define robustness as the minimal perturbation required to flip the predicted labels, which is applied in several adversarial attack studies (Moosavi-Dezfooli et al., 2016; Carlini and Wagner, 2017).

We then try to describe how the robustness-accuracy trade-off stems from the previously defined robust error. Recall that in supervised learning, we have a joint data distribution $p_d(x, y)$, and a discriminative model $p_\theta(y|x)$ for the label y, which is conditional on the input x. In standard settings, we obtain an accurate model via minimizing the expected KL divergence between $p_d(y|x)$ and $p_\theta(y|x)$, i.e., the **standard error** $\mathbf{R}_{\text{Standard}}(\theta)$ (see Eq. (1)) w.r.t. the parameters θ. The optimal solution θ^* satisfies $p_{\theta^*}(y|x) = p_d(y|x)$.

In adversarial settings, Madry et al. (2018) propose to minimize the **robust error** $\mathbf{R}_{\text{Madry}}(\theta)$ (see Eq. (3)) for training reliable models. Compared to the standard error, the robust error contains an inner maximization problem, finding the point $x' \in B(x)$ that maximizes the KL divergence between $p_d(y|x)$ and $p_\theta(y|x')$, where $B(x)$ is a set of allowed points around x. Minimizing the robust error w.r.t. θ imposes an inductive bias towards local invariance: for $\forall x' \in B(x)$, $p_\theta(y|x')$ is encouraged to be equal to $p_d(y|x)$. This locally-invariant bias makes the learned model tend to be over-smoothed, as observed in previous works (Stutz et al., 2020; Chen et al., 2021b). Hence, generally $p_{\theta^*}(y|x) \neq p_d(y|x)$ for the optimal θ^* of $\min_\theta \mathbf{R}_{\text{Madry}}(\theta)$. In Fig. 1, we observe that $p_{\theta^*}(y|x)$ does not converge to $p_d(y|x)$ even on

[1] Dept. of Comp. Sci. and Tech., Institute for AI, BNRist Center, THBI Lab, Tsinghua-Bosch Joint Center for ML, Tsinghua University. [2] Sea AI Lab, Singapore. Correspondence to: Jun Zhu <dcszj@tsinghua.edu.cn>, Shuicheng Yan <yansc@sea.com>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022. Copyright 2022 by the author(s).

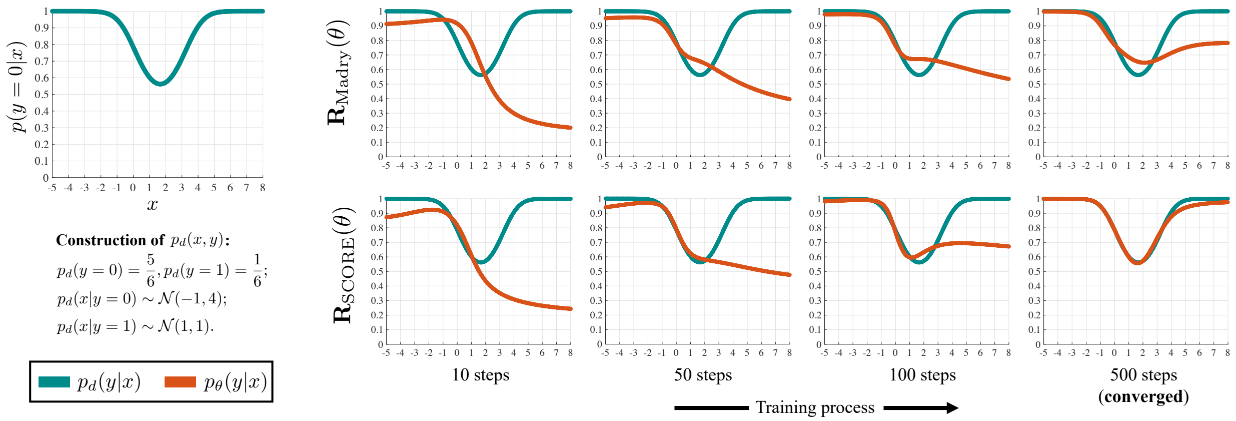

Figure 1. Toy demo on training $p_\theta(y|x)$ with $\mathbf{R}_{\text{Madry}}(\theta)$ and $\mathbf{R}_{\text{SCORE}}(\theta)$, respectively. We consider 1-d binary classification where $x \in \mathbb{R}$ and $y \in \{0, 1\}$, and the perturbed set $B(x) = \{x' \big| |x' - x| \leq 1\}$. Totally 60,000 training input-label pairs are sampled from the data distribution $p_d(x, y)$, as detailed in the left panel. The model $p_\theta(y|x)$ is a shallow MLP with two hidden layers and a sigmoid output. We use Adam optimizer and train for 500 steps to ensure convergence. The small gap between the converged $p_\theta(y|x)$ and $p_d(y|x)$ in the SCORE case is caused by limited model capacity. Note that in toy demo we assume $p_d(y|x)$ is accessible, which is not true in practice.

toy examples. The ending that $p_d(y|x)$ is not an optimally robust model w.r.t. $p_d(y|x)$ itself seems counter-intuitive, revealing that $\mathbf{R}_{\text{Madry}}(\theta)$ inherently does not support the reconciliation between robustness and accuracy.

Formally resolving the above inconsistency motivates us to properly redefine the robust error. To be specific, we substitute the inductive bias of local invariance with local equivariance, i.e., $\forall x' \in B(x)$, $p_\theta(y|x')$ is encouraged to point-wisely stick to $p_d(y|x')$, leading to the definition of **Self-COnsistent Robust Error (SCORE)** as $\mathbf{R}_{\text{SCORE}}(\theta)$ (see Eq. (4)). Compared to the robust error, SCORE finds the point x' that maximizes the KL divergence between $p_d(y|x')$ and $p_\theta(y|x')$ in its inner problem. SCORE aligns the optimal solution $p_{\theta^*}(y|x)$ with $p_d(y|x)$ (i.e., self-consistency, as shown in Fig. 1), while keeping the paradigm of robust optimization (Wald, 1945) in the finite-sample cases, as demonstrated in Fig. 2. More details can be found in Sec. 2.

How to optimize SCORE? Note that we only have closed-form access to $p_d(y|x)$ in toy cases. In practice, minimizing SCORE by the off-the-shelf first-order optimizers (e.g., SGD or Adam) requires estimating $\nabla_x \log p_d(y|x)$ (see Appendix B.2). This task can be decomposed into score matching tasks (Vincent, 2011) estimating the data scores $\nabla_x \log p_d(x)$ and $\nabla_x \log p_d(x|y)$, respectively. However, our initial experiments show that the estimated data scores are of high variance, making it non-trivial to well adopt them in the discriminative learning process. Fortunately, as described in Sec. 3, we find that by replacing KL divergence with any metric $\mathcal{D}$ satisfying the distance axioms (symmetry, triangle inequality), we can derive upper and lower bounds for distance-based SCORE $\mathbf{R}^{\mathcal{D}}_{\text{SCORE}}(\theta)$, and minimize it without knowing $\nabla_x \log p_d(y|x)$.

In Sec. 4, we bridge the gap between distance-based SCORE $\mathbf{R}^{\mathcal{D}}_{\text{SCORE}}(\theta)$ and previously used KL-based objectives like $\mathbf{R}_{\text{Madry}}(\theta)$ via Pinsker's inequality. This connection inspires us to propose instructive explanations for some well-known phenomena of robust models, e.g., overfitting (Rice et al., 2020) and semantic gradients (Ilyas et al., 2019).

In Sec. 5, we validate the effectiveness of replacing KL divergence with distance-based metrics (and their variants), developed from the analyses of SCORE. We improve the state-of-the-art AT methods under AutoAttack (Croce and Hein, 2020), and achieve top-rank performance with 1M DDPM generated data on the leader boards of CIFAR-10 and CIFAR-100 on RobustBench (Croce et al., 2020).

2. Self-Consistent Robust Error

According to Madry et al. (2018), robustness connects to a certain definition of robust error, which means a more robust model is a one that achieves lower robust error.[1] In this section, we first revisit previous definitions of robustness, and then propose a self-consistent robust error.

2.1. Preliminaries

In supervised learning, a training set $\{(x^i, y^i)\}_{i=1}^N$ consists of N i.i.d. input-label pairs (x^i, y^i) sampled from the joint data distribution $p_d(x, y)$. Let $p_\theta(y|x)$ be a discriminative model parameterized by θ. It is trained to match $p_d(y|x)$ by minimizing the standard error:

$$\mathbf{R}_{\text{Standard}}(\theta) = \mathbb{E}_{p_d(x)}\left[\text{KL}\left(p_d(y|x) \big\| p_\theta(y|x)\right)\right], \quad (1)$$

[1] Here, robustness refers to its differentiable surrogate form. In Sec. 5.1 we will discuss its 0-1 definition used for evaluation.

where $\text{KL}(P\|Q)$ denotes the KL divergence between two distributions P and Q. Since the data distribution p_d is independent of θ, minimizing $\mathbf{R}_{\text{Standard}}(\theta)$ w.r.t. θ is equivalent to minimizing the cross-entropy loss (Friedman et al., 2001), where the optimal solution is $p_{\theta^*}(y|x) = p_d(y|x)$.

2.2. Definitions of Robustness

There are several definitions of adversarial robustness in literature. For example, the seminal work of Szegedy et al. (2014) defines robustness as the minimal perturbation required to flip the predicted labels:

$$\mathbf{R}(x, \theta) = \min_{\delta} \|\delta\|, \text{ s.t. } \mathcal{Y}_\theta(x+\delta) \neq \mathcal{Y}_\theta(x), \tag{2}$$

where $\mathcal{Y}_\theta(x) = \arg\max_y p_\theta(y|x)$. Following this definition, several adversarial attacks are developed, aiming to find the successful evasions with minimal norms (Carlini and Wagner, 2017; Brendel et al., 2019; Rony et al., 2019; Pintor et al., 2021); margin-based defenses are also proposed (Tsuzuku et al., 2018; Pang et al., 2018; Ding et al., 2020). But solving the problem in Eq. (2) is computationally expensive, making it an intractable objective for end-to-end training. Thus, it is less discussed in the defense literature.

In contrast, Madry et al. (2018) propose PGD-AT and define robustness in terms of the robust error that is well compatible with supervised learning, formulated as

$$\mathbf{R}_{\text{Madry}}(\theta) = \mathbb{E}_{p_d(x)}\left[\max_{x' \in B(x)} \text{KL}\left(p_d(y|x) \| p_\theta(y|x')\right)\right]. \tag{3}$$

This definition is the most commonly used one, especially for adversarial training (AT) (Kannan et al., 2018; Wong et al., 2020; Shafahi et al., 2019; Rice et al., 2020). Notice that the original definition in Madry et al. (2018) is formulated by the cross-entropy loss, while Eq. (3) can be regarded as its expected version using KL divergence. In Appendix B.1, we show that these two versions are equivalent under first-order optimization.

Note that for $\forall x' \in B(x)$, $\mathbf{R}_{\text{Madry}}(\theta)$ encourages $p_\theta(y|x')$ to be locally invariant and equal to $p_d(y|x)$. As demonstrated in Fig. 1, generally there is $p_{\theta^*}(y|x) \neq p_d(y|x)$ when minimizing $\mathbf{R}_{\text{Madry}}(\theta)$. It is non-trivial to derive a closed-form solution for $p_{\theta^*}(y|x)$ even if we set $p_d(y|x)$ to be simple toy distributions. Previous works observe that the model learned by minimizing $\mathbf{R}_{\text{Madry}}(\theta)$ and its variants like TRADES (Zhang et al., 2019) will lead to over-smoothed decision landscapes (Stutz et al., 2020; Chen et al., 2021b).

2.3. A Self-Consistent Robust Error

If we admit $\mathbf{R}_{\text{Madry}}(\theta)$ as the proper definition of robust error, we would arrive at a paradox that $p_d(y|x)$ is not an optimally robust model w.r.t. $p_d(y|x)$ itself, which contradicts the basic preconditions of supervised learning.

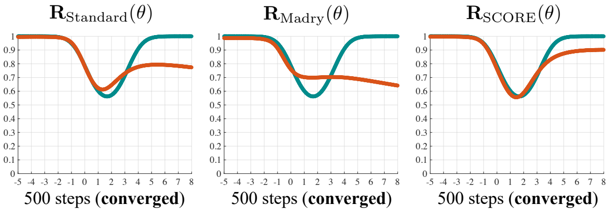

Figure 2. Note that Fig. 1 samples 60,000 training pairs, which suffice to well approximate the expectation $\mathbb{E}_{p_d(x)}$ in a 1-d toy problem. In contrast, here we only use 6 training pairs to mimic the finite-sample cases encountered in practice. We plot the converged states after 500 training steps for $\mathbf{R}_{\text{Standard}}(\theta)$, $\mathbf{R}_{\text{Madry}}(\theta)$, and $\mathbf{R}_{\text{SCORE}}(\theta)$, respectively. As can be seen, training with $\mathbf{R}_{\text{SCORE}}(\theta)$ inherits the benefits of robust optimization with a self-consistent optimality, which improves the sample efficiency of learning.

Thus, we suppose the misalignment between $p_{\theta^*}(y|x)$ and $p_d(y|x)$ when minimizing $\mathbf{R}_{\text{Madry}}(\theta)$ is one of the essential reasons for the trade-off between robustness and accuracy. To eliminate this trade-off, we slightly modify the definition of robust error and propose the **Self-COnsistent Robust Error (SCORE)**, which is formulated as

$$\mathbf{R}_{\text{SCORE}}(\theta) = \mathbb{E}_{p_d(x)}\left[\max_{x' \in B(x)} \text{KL}\left(p_d(y|x') \| p_\theta(y|x')\right)\right]. \tag{4}$$

Upon the inductive bias of local invariance imposed by $\mathbf{R}_{\text{Madry}}(\theta)$, SCORE makes amendment with local equivariance, allowing $p_\theta(y|x')$ to point-wisely match $p_d(y|x')$ for any $x' \in B(x)$. The self-consistency is thus achieved, or namely the optimal solution for minimizing $\mathbf{R}_{\text{SCORE}}(\theta)$ w.r.t. θ is $p_{\theta^*}(y|x) = p_d(y|x)$. In finite-sample cases, $\mathbf{R}_{\text{SCORE}}(\theta)$ preserves the paradigm of robust optimization (Wald, 1945) to cover the worst-case uncertainty, extracting more information from the samples in hand as seen in Fig. 2. The effect of promoting sample efficiency via robust optimization is also observed on large-scale tasks (Xie et al., 2020).

3. How to Practically Optimize SCORE?

While SCORE seems a promising objective, particularly for adversarial training, it is intractable to directly optimize $\mathbf{R}_{\text{SCORE}}(\theta)$ in practice. That is, minimizing $\mathbf{R}_{\text{SCORE}}(\theta)$ with existing first-order optimizers (e.g., SGD or Adam) requires closed-form access to $\nabla_x \log p_d(y|x)$ (see Appendix B.2). Although we can obtain training samples from $p_d(x, y)$, we cannot differentiate through the real data distribution. To this end, generative methods like score matching may be applied to estimate $\nabla_x \log p_d(y|x)$ (Vincent, 2011).

Nevertheless, through initial experiments we find that the estimated data scores are of high variance, and as a result, it is non-trivial to adopt them in the discriminative learning process. In this section, we elaborate on how to subtly avoid the need of directly optimizing $\mathbf{R}_{\text{SCORE}}(\theta)$ via resorting to distance metrics. Proofs of Theorems are in Appendix A.

3.1. Substituting KL Divergence with Distance Metrics

The KL divergence is *not* a distance metric, since it is asymmetric and does not satisfy the triangle inequality (Treves, 2016). In contrast, a distance metric $\mathcal{D}(\cdot\|\cdot)$ satisfies the three axioms of identity of indiscernibles, symmetry and the triangle inequality: $\mathcal{D}(A\|B) \leq \mathcal{D}(A\|C) + \mathcal{D}(C\|B)$. Typical examples include ℓ_p-distances for $p \geq 1$, where $\|A-B\|_p \leq \|A-C\|_p + \|C-B\|_p$. By substituting KL divergence with a certain distance metric $\mathcal{D}(\cdot\|\cdot)$, we denote

$$\mathbf{R}^{\mathcal{D}}_{\text{Madry}}(\theta) = \mathbb{E}_{p_d(x)}\left[\max_{x'\in B(x)} \mathcal{D}\left(p_d(y|x)\|p_\theta(y|x')\right)\right]; \quad (5)$$

$$\mathbf{R}^{\mathcal{D}}_{\text{SCORE}}(\theta) = \mathbb{E}_{p_d(x)}\left[\max_{x'\in B(x)} \mathcal{D}\left(p_d(y|x')\|p_\theta(y|x')\right)\right]. \quad (6)$$

Intriguingly, now we can derive upper and lower bounds for $\mathbf{R}^{\mathcal{D}}_{\text{SCORE}}(\theta)$ using $\mathbf{R}^{\mathcal{D}}_{\text{Madry}}(\theta)$, as stated below:

Theorem 1. *(Bounding the SCORE objective) For any distance metric $\mathcal{D}(\cdot\|\cdot)$, there are lower and upper bounds that*

$$|\mathbf{R}^{\mathcal{D}}_{\text{Madry}}(\theta) - C^{\mathcal{D}}| \leq \mathbf{R}^{\mathcal{D}}_{\text{SCORE}}(\theta) \leq \mathbf{R}^{\mathcal{D}}_{\text{Madry}}(\theta) + C^{\mathcal{D}},$$

$$\text{where } C^{\mathcal{D}} = \mathbb{E}_{p_d(x)}\left[\max_{x'\in B(x)} \mathcal{D}\left(p_d(y|x)\|p_d(y|x')\right)\right]$$

is a constant independent of θ. The lower bound becomes tight when $\mathbf{R}^{\mathcal{D}}_{\text{SCORE}}(\theta) = 0$, and we have $\mathbf{R}^{\mathcal{D}}_{\text{Madry}}(\theta) = C^{\mathcal{D}}$.

What is the constant $C^{\mathcal{D}}$? The constant $C^{\mathcal{D}}$ indicates the intrinsic smoothness of the data distribution, i.e., how much $p_d(y|x)$ has changed in the neighborhood set $B(x)$. Though we cannot precisely compute $C^{\mathcal{D}}$ in practice, more complex datasets are expected to have larger values of $C^{\mathcal{D}}$.

Upper bound. $\mathbf{R}^{\mathcal{D}}_{\text{SCORE}}(\theta) \leq \mathbf{R}^{\mathcal{D}}_{\text{Madry}}(\theta) + C^{\mathcal{D}}$ guarantees that we can relax the goal from minimizing $\mathbf{R}^{\mathcal{D}}_{\text{SCORE}}(\theta)$ to minimizing $\mathbf{R}^{\mathcal{D}}_{\text{Madry}}(\theta)$, without the need to estimate data scores. A similar trick of optimizing upper bounds is widely applied in variational learning (Kingma and Welling, 2014).

Lower bound. $|\mathbf{R}^{\mathcal{D}}_{\text{Madry}}(\theta) - C^{\mathcal{D}}| \leq \mathbf{R}^{\mathcal{D}}_{\text{SCORE}}(\theta)$ tells us that: **(i)** $\mathbf{R}^{\mathcal{D}}_{\text{Madry}}(\theta) = C^{\mathcal{D}}$ is a necessary condition for $\mathbf{R}^{\mathcal{D}}_{\text{SCORE}}(\theta) = 0$; **(ii)** overly minimizing $\mathbf{R}^{\mathcal{D}}_{\text{Madry}}(\theta)$ to approach zero makes $\mathbf{R}^{\mathcal{D}}_{\text{SCORE}}(\theta)$ tend to increase or overfit. In Sec. 4.1, we will show that a similar conclusion holds for the original KL-based $\mathbf{R}_{\text{Madry}}(\theta)$, which closely connects to the overfitting phenomenon observed in Rice et al. (2020).

A toy demo. In Fig. 3, we leverage a toy example to further explain the above properties claimed in Theorem 1. During training, we minimize $\mathbf{R}^{\ell_2}_{\text{Madry}}(\theta)$, and the constant C^{ℓ_2} can be computed in closed-form. As can be seen, in the initial phase of training (first ~ 100 training steps), the upper bound works, and $\mathbf{R}^{\ell_2}_{\text{SCORE}}(\theta)$ is effectively minimized.

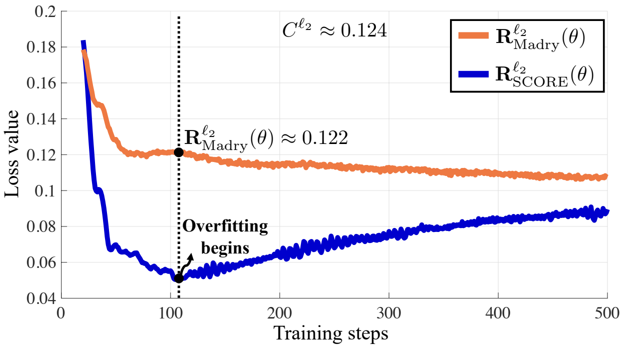

Figure 3. The overfitting phenomenon encountered when minimizing $\mathbf{R}^{\mathcal{D}}_{\text{Madry}}(\theta)$. We use the same toy data distribution as in Fig. 1, and set $\mathcal{D}$ to be ℓ_2-distance. As annotated, when $\mathbf{R}^{\ell_2}_{\text{Madry}}(\theta) \approx C^{\ell_2}$ (≈ 0.12 in this example), the overfitting phenomenon happens, i.e., $\mathbf{R}^{\ell_2}_{\text{SCORE}}(\theta)$ begins to increase while $\mathbf{R}^{\ell_2}_{\text{Madry}}(\theta)$ still decreases.

Then after $\mathbf{R}^{\ell_2}_{\text{Madry}}(\theta)$ is minimized to be less than C^{ℓ_2}, the lower bound intervenes and overfitting happens. Intuitively, when $\mathbf{R}^{\mathcal{D}}_{\text{Madry}}(\theta) < C^{\mathcal{D}}$, the learned model $p_\theta(y|x)$ is considered as over-smoothed compared to the oracle $p_d(y|x)$.

3.2. Monotonically Increasing Convex Variants

In the above, we show that substituting KL divergence with any distance metric $\mathcal{D}$ can induce favorable bounds, which enable us to optimize SCORE efficiently and forebode the overfitting phenomenon. However, the KL divergence involves logarithm function like $\log p_\theta(y|x)$, whose gradients may focus on unlearned examples (with small values of $p_\theta(y|x)$). In contrast, a distance metric is sublinear and does not work well in practice, as empirically shown in Table 1. Thus, we generalize Theorem 1 to monotonically increasing convex variants of $\mathcal{D}$, formalized as below:

Theorem 2. *(Variants of $\mathcal{D}$) For any distance metric $\mathcal{D}(\cdot\|\cdot)$ and a monotonically increasing convex function $\phi(\cdot)$,*

$$|\mathbf{R}^{\mathcal{D}}_{\text{SCORE}}(\theta) - C^{\mathcal{D}}| \leq \phi^{-1}\left(\mathbf{R}^{\phi\circ\mathcal{D}}_{\text{Madry}}(\theta)\right),$$

where $\phi^{-1}(\cdot)$ is the inverse function, and the superscript $\phi\circ\mathcal{D}$ means using the composition $\phi\circ\mathcal{D}$ in Eq. (5).

Remark. Theorem 2 allows us to construct upper bounds of $\mathbf{R}^{\mathcal{D}}_{\text{SCORE}}(\theta)$ utilizing more general variants of distance, e.g., squared error (SE) $\|P-Q\|_2^2$ or JS divergence $\text{JS}(P\|Q)$, based on the fact that $\sqrt{\text{JS}(P\|Q)}$ is a distance metric (Endres and Schindelin, 2003). These variants work much better than their distance counterparts, as seen in Table 1.

In our experiments, we take SE as an example derived from our analyses, and verify that substituting KL divergence with SE in training objectives improves the performance of state-of-the-art AT methods, as will be detailed in Sec. 5.

3.3. Equivalent Relation Induced by Distance Metrics

Besides PGD-AT (Madry et al., 2018), another typical AT method is TRADES (Zhang et al., 2019). Given any distance metric $\mathcal{D}$, $\mathbf{R}^{\mathcal{D}}_{\text{Madry}}(\theta)$ and $\mathbf{R}^{\mathcal{D}}_{\text{TRADES}}(\theta;\beta)$ (defined in Eq. (21)) are equivalent in the parameter space, i.e., induce the same topology of loss landscapes (Conrad, 2018):

Theorem 3. *(Equivalent Relation) For any distance metric $\mathcal{D}(\cdot||\cdot)$ and a given hyperparameter $\beta \geq 1$, there is*

$$\mathbf{R}^{\mathcal{D}}_{\text{Madry}}(\theta) \leq \mathbf{R}^{\mathcal{D}}_{\text{TRADES}}(\theta;\beta) \leq (1+2\beta)\cdot\mathbf{R}^{\mathcal{D}}_{\text{Madry}}(\theta),$$

which holds for any model parameters θ.

Therefore, the conclusions that hold for $\mathbf{R}^{\mathcal{D}}_{\text{Madry}}(\theta)$ similarly hold for $\mathbf{R}^{\mathcal{D}}_{\text{TRADES}}(\theta;\beta)$, as justified in our experiments.

4. New Insights Brought by SCORE

Although KL divergence is not a distance metric, Pinsker's inequality (Csiszár and Körner, 2011) claims that

$$\frac{1}{2}\|P-Q\|_1^2 \leq \text{KL}(P||Q) \tag{7}$$

holds for two distributions P and Q. This connects KL divergence with distance metrics, leading to some interesting views for explaining previous observations on robust models, like overfitting and semantic input gradients. We would not say that our explanations are conclusive; instead, we aim to provide insights to explore these phenomena further.

4.1. Overfitting and Early-Stopping

Rice et al. (2020) observe that overfitting happens in the process of adversarial training, and simply using early-stopping can reduce the robust generalization gap. Previous works attribute the overfitting phenomenon to activation functions (Singla et al., 2021), perturbation underfitting (Li et al., 2020b) or hard instances (Liu et al., 2021). In contrast, we find a more straightforward explanation from the view of SCORE. Specifically, according to Theorem 2 and Pinsker's inequality, there is

Corollary 1. *Let ℓ_1 refers to ℓ_1-distance metric. There is*

$$|\mathbf{R}^{\ell_1}_{\text{SCORE}}(\theta) - C^{\ell_1}| \leq \sqrt{2\cdot\mathbf{R}_{\text{Madry}}(\theta)}.$$

The square-root robust error $\sqrt{2\cdot\mathbf{R}_{\text{Madry}}(\theta)}$ acts as an upper bound for the SCORE $\mathbf{R}^{\ell_1}_{\text{SCORE}}(\theta)$ in the initial phase of training (i.e., when $\mathbf{R}^{\ell_1}_{\text{SCORE}}(\theta) > C^{\ell_1}$), which coincides with the observation that accuracy and robustness can both increase before overfitting happens (Rice et al., 2020). As can be seen, the necessary condition for $\mathbf{R}^{\ell_1}_{\text{SCORE}}(\theta)=0$ is

$$C^{\ell_1} \leq \sqrt{2\cdot\mathbf{R}_{\text{Madry}}(\theta)} \Longrightarrow \mathbf{R}_{\text{Madry}}(\theta) \geq \frac{\left(C^{\ell_1}\right)^2}{2}, \tag{8}$$

which implies $\mathbf{R}_{\text{Madry}}(\theta)$ should be early-stopped at least before $\left(C^{\ell_1}\right)^2/2$ to avoid the overfitting phenomenon. In Appendix D.1, we show that the overfitting actually happens even earlier at $\mathbf{R}_{\text{Madry}}(\theta) \approx C^{\text{KL}}$ (demo in Fig. 6 (a)).

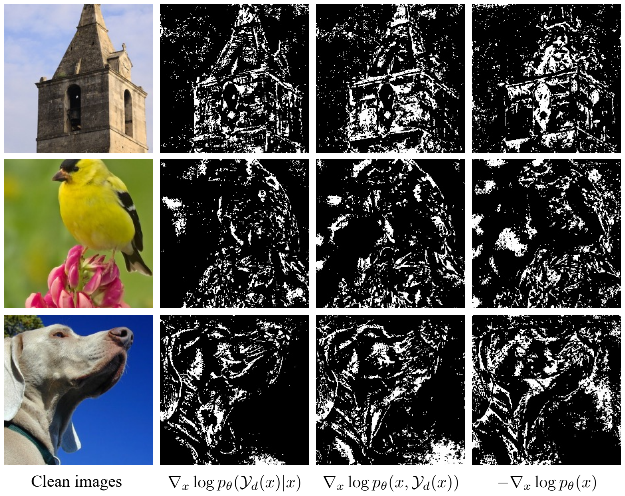
Clean images
$\nabla_x \log p_\theta(\mathcal{Y}_d(x)|x)$
$\nabla_x \log p_\theta(x, \mathcal{Y}_d(x))$
$-\nabla_x \log p_\theta(x)$

Figure 4. Visualization of semantic gradients. We adopt a ResNet-50 model adversarially trained by FreeAT on ImageNet. Here $\nabla_x \log p_\theta(x, \mathcal{Y}_d(x))$ and $-\nabla_x \log p_\theta(x)$ are constructed according to Grathwohl et al. (2020) (details in Appendix D.2), and we set $\mathcal{Y}_d(x)$ to be the test label of each clean image. We add up the partial derivatives of three RGB channels for each pixel position, and sort out the top 10% pixel positions with large values of total derivatives (i.e., those affect the objectives the most) in the plots.

4.2. Semantic Gradients: Adversarial Training

Previous studies observe that the input gradients $\nabla_x p_\theta(y|x)$ of the adversarially trained models exhibit semantic or perceptually-aligned characteristics (Tao et al., 2018; Ilyas et al., 2019; Santurkar et al., 2019; Etmann et al., 2019; Chan et al., 2020). Recall that minimizing $\mathbf{R}_{\text{Madry}}(\theta)$ effectively minimizes the gap between $\mathbf{R}^{\ell_1}_{\text{SCORE}}(\theta)$ and C^{ℓ_1}. Considering this, in the ℓ_p-norm threat model of $B(x) = \{x' | \|x'-x\|_p \leq \epsilon\}$, we take the first-order expansion similar as in Simon-Gabriel et al. (2019), and obtain

Theorem 4. *(Gradient Alignment) Assume that in training, there is $p_d(y|x) > p_\theta(y|x)$ for $y = \mathcal{Y}_d(x)$; otherwise $p_d(y|x) < p_\theta(y|x)$. Then under first-order expansion,*

$$\mathbf{R}^{\ell_1}_{\text{SCORE}}(\theta) = \mathbf{R}^{\ell_1}_{\text{Standard}}(\theta) + \\ 2\epsilon\cdot\mathbb{E}_{p_d(x)}\left[\|\nabla_x p_d(\mathcal{Y}_d(x)|x) - \nabla_x p_\theta(\mathcal{Y}_d(x)|x)\|_q\right] + o(\epsilon),$$

where $\mathbf{R}^{\ell_1}_{\text{Standard}}(\theta)$ is the standard error using ℓ_1-distance, $\mathcal{Y}_d(x) = \arg\max_y p_d(y|x)$, and $\|\cdot\|_q$ is the dual of $\|\cdot\|_p$.

The assumption in Theorem 4 is trivially true if we substitute $p_d(y|x)$ with a one-hot vector, as in supervised learning. Then, minimizing $\mathbf{R}_{\text{Madry}}(\theta)$ (and thus $\mathbf{R}^{\ell_1}_{\text{SCORE}}(\theta)$) encourages gradient alignment between $\nabla_x p_d(\mathcal{Y}_d(x)|x)$ and $\nabla_x p_\theta(\mathcal{Y}_d(x)|x)$ (demo in Fig. 6 (b)). This is analogous to score matching, but is not exactly the same since $p(y|x)$ is not normalized w.r.t. x, i.e., $\int_x p(y|x)dx \neq 1$. Still, adversarial training enjoys generative learning patterns.

In Fig. 4, we use a ResNet-50 trained by FreeAT (Shafahi et al., 2019) on ImageNet, and visualize its input gradients $\nabla_x \log p_\theta(\mathcal{Y}_d(x)|x)$. We also plot $\nabla_x \log p_\theta(x, \mathcal{Y}_d(x))$ and $\nabla_x \log p_\theta(x)$ via regarding the classifier as an implicit EBM (Grathwohl et al., 2020). We highlight prominent pixels with large derivatives. As seen, adversarially trained models catch shape-based features, which are more generative compared to texture-based ones (Geirhos et al., 2019).

4.3. Semantic Gradients: Randomized Smoothing

Randomized smoothing is a promising method towards scalable certified defenses (Cohen et al., 2019). The smoothed model is trained by Gaussian augmentation and returns an ensemble prediction. Inspired by Theorem 4, we discover a similar gradient alignment objective in randomized smoothing, coincident with the empirical report that randomized smoothing leads to semantic gradients (Kaur et al., 2019).

Specifically, let $p_d^\sigma(x,y)$ be the data distribution augmented by a zero-mean Gaussian noise denoted as $\sqrt{\sigma}\omega$, where $\omega \sim \mathcal{N}(0,I)$ is a standard Gaussian and $p_d^\sigma(x,y)$ can be written as $p_d^\sigma(x,y) = \mathbb{E}_{\mathcal{N}(\omega;0,I)}\left[p_d(x-\sqrt{\sigma}\omega, y)\right]$. We consider the Gaussian-augmented cross-entropy loss[2]

$$\mathbf{R}_G(\theta;\sigma) = \mathbb{E}_{p_d^\sigma(x,y)}\left[-\log p_\theta(y|x)\right], \tag{9}$$

where $\mathbf{R}_G(\theta;0)$ degenerates to the cross-entropy loss. Then we derive the loss derivative of $\mathbf{R}_G(\theta;\sigma)$ w.r.t. σ as

Theorem 5. *(Gradient Alignment) Given any model parameters θ, the loss derivative of $\mathbf{R}_G(\theta;\sigma)$ w.r.t. σ is*

$$\frac{d}{d\sigma}\mathbf{R}_G(\theta;\sigma) = \frac{1}{2}\mathbb{E}_{p_d^\sigma(x,y)}\left[\nabla_x \log p_\theta(y|x)^\top \nabla_x \log p_d^\sigma(x|y)\right]$$

where $p_d^\sigma(x|y)$ is defined as $\mathbb{E}_{\mathcal{N}(\omega;0,I)}\left[p_d(x-\sqrt{\sigma}\omega|y)\right]$.

In short, Theorem 5 is proved following a similar routine as in Lyu (2009). Particularly, for small values of σ, there is

$$\mathbf{R}_G(\theta;\sigma) = \mathbf{R}_G(\theta;0) + \sigma \cdot \frac{d}{d\sigma}\mathbf{R}_G(\theta;\sigma)\Big|_{\sigma=0} + o(\sigma), \tag{10}$$

which says that Gaussian-augmented learning could be decomposed into standard learning (first term), and gradient alignment (second term) encouraging $\nabla_x \log p_\theta(y|x)$ to match the direction of $-\nabla_x \log p_d(x|y)$. This result adds to the evidence that robust learning has generative properties.

Table 1. Classification accuracy (%) on clean images and under 10-steps PGD attack. Here we use a ResNet-18 model trained on CIFAR-10. We do not test under AutoAttack since this is only a qualitative study on effects of different losses used in PGD-AT. We find distance metrics do not work well in practice, while different losses adopt different suitable learning rates as highlighted.

Loss	Alias	$l.r. = 0.1$		$l.r. = 0.05$		$l.r. = 0.01$	
		Clean	PGD	Clean	PGD	Clean	PGD
$\|P-Q\|_2$	ℓ_2-dis.	75.91	52.16	77.98	52.74	78.45	51.13
$\|P-Q\|_1$	ℓ_1-dis.	58.51	43.87	64.88	46.77	70.02	47.76
$\|P-Q\|_\infty$	ℓ_∞-dis.	58.34	43.71	59.75	45.02	65.65	46.36
$\sqrt{\text{JS}(P\|Q)}$	JS-dis.	53.06	40.08	55.27	41.86	68.50	46.49
$\text{JS}(P\|Q)$	JS-div.	79.41	51.75	81.27	51.85	80.12	49.10
$\text{KL}(P\|Q)$	KL-div.	82.74	53.02	83.21	51.52	82.65	47.45
$\|P-Q\|_1^2$	-	79.87	50.96	81.49	52.00	81.26	47.51
$\|P-Q\|_2^2$	SE	80.59	54.63	83.38	54.01	81.43	51.13

5. Experiments

In this section we first discuss the 0-1 version of SCORE used for evaluation, to show the criterion is aligned with common practice; then, we demonstrate empirical results. Code is at https://github.com/P2333/SCORE.

5.1. The 0-1 Version of SCORE for Evaluation

In the test phase, we mostly apply 0-1 version of errors for evaluation (Friedman et al., 2001). Specifically, the 0-1 version of the standard error $\mathbf{R}_{\text{Standard}}(\theta)$ can be written as

$$\mathbb{E}_{p_d(x)}\left[\mathbf{1}\left(\mathcal{Y}_\theta(x) \neq \mathcal{Y}_d(x)\right)\right]. \tag{11}$$

Recall that $\mathcal{Y}_\theta(x)$ and $\mathcal{Y}_d(x)$ are the hard labels obtained by taking $\arg\max$ w.r.t. $p_\theta(y|x)$ and $p_d(y|x)$, respectively. Similarly, the 0-1 version of the robust error $\mathbf{R}_{\text{Madry}}(\theta)$ is

$$\mathbb{E}_{p_d(x)}\left[\max_{x' \in B(x)} \mathbf{1}\left(\mathcal{Y}_\theta(x') \neq \mathcal{Y}_d(x)\right)\right]. \tag{12}$$

Then one would naturally wonder: does SCORE induce a new evaluation criterion? Formally, the 0-1 version of $\mathbf{R}_{\text{SCORE}}(\theta)$ is formulated as

$$\mathbb{E}_{p_d(x)}\left[\max_{x' \in B(x)} \mathbf{1}\left(\mathcal{Y}_\theta(x') \neq \mathcal{Y}_d(x')\right)\right]. \tag{13}$$

Although SCORE advocates that $p_d(y|x')$ changes w.r.t. x', the hard label $\mathcal{Y}_d(x')$ can be reasonably assumed to be invariant, i.e., $\mathcal{Y}_d(x') = \mathcal{Y}_d(x)$ for any $x' \in B(x)$. Actually this assumption is widely accepted in the adversarial community (Biggio et al., 2013; Goodfellow et al., 2015). Given this, we conclude that the criteria in Eq. (12) and Eq. (13) are equivalent, and then $p_{\theta^*}(y|x) = p_d(y|x)$ is the optimal solution for both of them. Thus in our experiments, *we do not need to modify the commonly used evaluation criterion.*

[2] We cannot write the loss in the form of KL divergence since there is $p_d^\sigma(x,y) \neq p_d^\sigma(x)p_d(y|x)$, as shown in Eq. (26).

Table 2. Classification accuracy (%) on clean images and under AutoAttack (ℓ_∞, $\epsilon = 8/255$). Here we use ResNet-18 trained by PGD-AT or TRADES on CIFAR-10, using KL divergence or squared error (SE) as the loss function. Clipping loss is executed at every training step, compatible with early-stopping. We average the results over five runs and report the mean ± standard deviation.

Method	Loss	Clip	Clean	AutoAttack
PGD-AT	KL div.	-	82.46 ± 0.41	48.39 ± 0.14
	SE	✗	82.13 ± 0.14	49.41 ± 0.27
	SE	✓	$\mathbf{82.80 \pm 0.16}$	$\mathbf{49.63 \pm 0.17}$
TRADES	KL div.	-	81.47 ± 0.12	49.14 ± 0.16
	SE	✗	83.50 ± 0.05	49.44 ± 0.35
	SE	✓	$\mathbf{83.75 \pm 0.14}$	$\mathbf{49.57 \pm 0.28}$

5.2. Basic Setting without Extra or Generated Data

In the basic setting, neither extra nor generated data are used. We follow Pang et al. (2021) and apply ResNet-18 (He et al., 2016) as the model architecture. In training, we use SGD momentum optimizer with batch size 128 and weight decay 5×10^{-4}. We exploit the PGD-AT (Madry et al., 2018) and TRADES (Zhang et al., 2019) frameworks. The training attack used is 10-steps PGD with step size $\alpha = 2/255$ for ℓ_∞ threat model and $\alpha = 16/255$ for ℓ_2 threat model. The training runs for 110 epochs with the learning rate decaying by a factor of 0.1 at the 100 and 105 epoch, respectively. The hyperparameter $\beta = 6$ in the TRADES experiments.

Distance metric does not work well. We first ablate on the effectiveness of different losses used in the PGD-AT objective $\mathbf{R}_{\text{Madry}}(\theta)$. As reported in Table 1, we choose ℓ_1-, ℓ_2-, ℓ_∞-, and JS distances, their squared variants, as well as KL divergence. We set three initial learning rates of 0.1, 0.05, and 0.01. We evaluate under PGD attacks (Madry et al., 2018) for a qualitative study. As can be seen, distance metrics do not work well in practice due to their sublinear property. Thus, we select squared error (SE) as a typical example developed by our analyses and compare effectiveness of SE-based instantiations with KL-based baselines.

SE outperforms KL divergence. In Table 2, we substitute KL divergence in the objectives of PGD-AT and TRADES with SE. According to Table 1, we use 0.05 initial learning rate for our methods and 0.1 for baselines. For our methods, we report the results on the checkpoint with the highest value of PGD-10 (SE) accuracy on a separate validation set, similarly to Rice et al. (2020). The best checkpoint is selected for baselines by the highest value of PGD-10 (KL). Besides, we also introduce a clipping operation to the loss values. For PGD-AT, we choose the clipping threshold of 0.4, and for TRADES, we choose 0.3. We evaluate the model robustness under AutoAttack (Croce and Hein, 2020). SE can improve clean accuracy and/or robustness for free, i.e., without extra computation.

Table 3. Classification accuracy (%) on clean images and under AutoAttack (ℓ_∞, $\epsilon = 8/255$). The model is WRN-28-10 (SiLU), following the training pipeline in Rebuffi et al. (2021) and using 1M DDPM generated data. KL divergence is substituted with the SE function in TRADES, and no clipping loss is executed.

Dataset	β	Clean	AutoAttack
CIFAR-10	6	86.64 ± 0.13	60.78 ± 0.16
	5	87.19 ± 0.20	61.05 ± 0.11
	4	87.89 ± 0.19	61.11 ± 0.27
	3	88.60 ± 0.13	60.89 ± 0.09
	2	89.28 ± 0.15	60.13 ± 0.21
CIFAR-100	4	61.94 ± 0.13	31.21 ± 0.12
	3	63.12 ± 0.37	31.01 ± 0.09

5.3. Advanced Setting with DDPM Generated Data

Recent progress shows that generative models trained solely on the original training set can be leveraged to improve model robustness (Rebuffi et al., 2021; Gowal et al., 2021; Sehwag et al., 2021). We thus follow the setting of Rebuffi et al. (2021) and its PyTorch implementation[3], using the provided 1M DDPM (Ho et al., 2020) generated data. For the sake of completeness, we briefly recap the training setup here. We use SiLU activation function (Hendrycks and Gimpel, 2016) with WideResNet (WRN) backbones (Zagoruyko and Komodakis, 2016). We adopt weight averaging (Izmailov et al., 2018) with $\tau = 0.995$. For optimizer we use SGD with Nesterov momentum (Nesterov, 1983), momentum factor being 0.9 and weight decay 5×10^{-4}. We further use cyclic learning rates (Smith and Topin, 2019) with cosine annealing. We use a batch size of 512 and train for 400 epochs (with an initial learning rate of 0.2) due to limited computational resources; a larger batch size of 1024 and longer training of 800 epochs (with an initial learning rate of 0.4) could further improve our method, as employed in Rebuffi et al. (2021). The original-to-generated data ratio in each batch is 0.3 on CIFAR-10 and 0.4 on CIFAR-100 (Gowal et al., 2021).

Values of β. In Table 3, we study the effect of β in TRADES on the robustness-accuracy trade-off. Unlike previous observations (Zhang et al., 2019), we find when using large models with extra generated data, smaller values of β (i.e., 3 or 4) facilitate higher clean accuracy while keeping robust accuracy under AutoAttack almost unchanged. Other more advanced ways for tuning β can also be applied (Wu et al., 2021). Theoretically, Theorem 3 tells us that smaller values of β make TRADES more aligned with PGD-AT.

Comparison with state-of-the-art. In Table 4, we consider threat models of (ℓ_∞, $\epsilon = 8/255$) and (ℓ_2, $\epsilon = 128/255$) on CIFAR-10, and (ℓ_∞, $\epsilon = 8/255$) on CIFAR-100. We resort to RobustBench (Croce et al., 2020) for comparing with the

[3] https://github.com/imrahulr/adversarial_robustness_pytorch

Table 4. Classification accuracy (%) on clean images and under AutoAttack. The results of our methods are in **bold**, and no clipping loss is executed. Here ‡ means *no CutMix applied*, following Rade and Moosavi-Dezfooli (2021). We use a batch size of 512 and train for 400 epochs due to limited resources, while a larger batch size of 1024 and training for 800 epochs are expected to achieve better performance.

Dataset	Method	Architecture	DDPM	Batch	Epoch	Clean	AutoAttack
CIFAR-10 (ℓ_∞, $\epsilon = 8/255$)	Rice et al. (2020)	WRN-34-20	✗	128	200	85.34	53.42
	Zhang et al. (2020)	WRN-34-10	✗	128	120	84.52	53.51
	Pang et al. (2021)	WRN-34-20	✗	128	110	86.43	54.39
	Wu et al. (2020)	WRN-34-10	✗	128	200	85.36	56.17
	Gowal et al. (2020)	WRN-70-16	✗	512	200	85.29	57.14
	Rebuffi et al. (2021)‡	WRN-28-10	1M	1024	800	85.97	60.73
	+ **Ours** (KL → SE, $\beta = 3$)	WRN-28-10	1M	512	400	**88.61**	**61.04**
	+ **Ours** (KL → SE, $\beta = 4$)	WRN-28-10	1M	512	400	**88.10**	**61.51**
	Rebuffi et al. (2021)‡	WRN-70-16	1M	1024	800	86.94	63.58
	+ **Ours** (KL → SE, $\beta = 3$)	WRN-70-16	1M	512	400	**89.01**	**63.35**
	+ **Ours** (KL → SE, $\beta = 4$)	WRN-70-16	1M	512	400	**88.57**	**63.74**
	Gowal et al. (2021)	WRN-70-16	100M	1024	2000	88.74	66.10
CIFAR-10 (ℓ_2, $\epsilon = 128/255$)	Wu et al. (2020)	WRN-34-10	✗	128	200	88.51	73.66
	Gowal et al. (2020)	WRN-70-16	✗	512	200	90.90	74.50
	Rebuffi et al. (2021)‡	WRN-28-10	1M	1024	800	90.24	77.37
	+ **Ours** (KL → SE, $\beta = 3$)	WRN-28-10	1M	512	400	**91.52**	**77.89**
	+ **Ours** (KL → SE, $\beta = 4$)	WRN-28-10	1M	512	400	**90.83**	**78.10**
CIFAR-100 (ℓ_∞, $\epsilon = 8/255$)	Wu et al. (2020)	WRN-34-10	✗	128	200	60.38	28.86
	Gowal et al. (2020)	WRN-70-16	✗	512	200	60.86	30.03
	Rebuffi et al. (2021)‡	WRN-28-10	1M	1024	800	59.18	30.81
	+ **Ours** (KL → SE, $\beta = 3$)	WRN-28-10	1M	512	400	**63.66**	**31.08**
	+ **Ours** (KL → SE, $\beta = 4$)	WRN-28-10	1M	512	400	**62.08**	**31.40**
	Rebuffi et al. (2021)‡	WRN-70-16	1M	1024	800	60.46	33.49
	+ **Ours** (KL → SE, $\beta = 3$)	WRN-70-16	1M	512	400	**65.56**	**33.05**
	+ **Ours** (KL → SE, $\beta = 4$)	WRN-70-16	1M	512	400	**63.99**	**33.65**

top-performance robust models. We run on the model architectures of WRN-28-10 and WRN-70-16. Note that we do not apply *CutMix* following (Rade and Moosavi-Dezfooli, 2021), since we find the effectiveness of CutMix may rely on the specific implementation of parallel computing and requires further exploration. As observed, simply substituting KL divergence with SE and using a relatively small value of β can significantly improve clean accuracy, with comparable or even better robustness.

6. Conclusion and Discussion

We attribute the trade-off between robustness and accuracy to the improper definition of robustness. Essentially, the robust error $\mathbf{R}_{\text{Madry}}(\theta)$ is a surrogate objective of its 0-1 form, but the problem is that it unintentionally converts the hard-label invariance (i.e., $\mathcal{Y}_d(x')$ is unchanged) into the distributional invariance (i.e., $p_d(y|x')$ should equal $p_d(y|x)$) for $\forall x' \in B(x)$. The former is a reasonable inductive bias, while the latter is an overcorrection towards smoothness. Thus we propose SCORE and provide an efficient way to optimize it. SCORE brings us many new insights for explaining the phenomena of overfitting and semantic gradients encountered on robust models. Inspired by SCORE, substituting KL divergence with SE effectively alleviates the empirical robustness-accuracy trade-off.

Empirical trade-off still exists. SCORE ensures that robustness and accuracy are reconcilable in the sense of taking expectation w.r.t. p_d. However, in the finite-sample cases, the insufficiency of training data could cause empirical trade-off (Schmidt et al., 2018). This effect is also observed in Fig. 2 when using the standard error or SCORE. Fortunately, SCORE promises that the trained model will finally converge to a self-consistent solution as more data are collected.

$B(x)$ **can be arbitrary in SCORE.** The local invariance assumed in $\mathbf{R}_{\text{Madry}}(\theta)$ constrains the allowed set $B(x)$ to be local around x (e.g., ℓ_p-balls). In contrast, SCORE exploits equivariance instead of invariance; thus, $B(x)$ could be an arbitrary set (e.g., involving the points with similar semantics as x), whereas most of the conclusions proved for SCORE still hold. Given this, we can apply SCORE to a wider range of tasks, with guarantees of self-consistency and enjoying sample efficiency brought by robust optimization.

Acknowledgements

The authors would like to thank Kun Xu for the insightful discussion on the idea of SCORE proposed in this paper. This work was supported by National Key Research and Development Program of China (Nos. 2020AAA0104304, 2020AAA0106302, 2017YFA0700904), NSFC Projects (Nos. 62061136001, 61621136008, 62076147, U19B2034, U19A2081, U1811461), Tsinghua-Huawei Joint Research Program, a grant from Tsinghua Institute for Guo Qiang, and the High Performance Computing Center, Tsinghua University. Tianyu Pang was partly supported by Microsoft Research Asia Fellowship and Baidu Scholarship.

References

Jean-Baptiste Alayrac, Jonathan Uesato, Po-Sen Huang, Alhussein Fawzi, Robert Stanforth, and Pushmeet Kohli. Are labels required for improving adversarial robustness? In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 12192–12202, 2019.

Maksym Andriushchenko and Nicolas Flammarion. Understanding and improving fast adversarial training. In *Advances in neural information processing systems (NeurIPS)*, 2020.

Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *International Conference on Machine Learning (ICML)*, 2018.

Yogesh Balaji, Tom Goldstein, and Judy Hoffman. Instance adaptive adversarial training: Improved accuracy tradeoffs in neural nets. *arXiv preprint arXiv:1910.08051*, 2019.

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 *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 387–402. Springer, 2013.

Wieland Brendel, Jonas Rauber, Matthias Kümmerer, Ivan Ustyuzhaninov, and Matthias Bethge. Accurate, reliable and fast robustness evaluation. *Advances in Neural Information Processing Systems*, 32:12861–12871, 2019.

Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *IEEE Symposium on Security and Privacy (S&P)*, 2017.

Nicholas Carlini, Anish Athalye, Nicolas Papernot, Wieland Brendel, Jonas Rauber, Dimitris Tsipras, Ian Goodfellow, Aleksander Madry, and Alexey Kurakin. On evaluating adversarial robustness. *arXiv preprint arXiv:1902.06705*, 2019.

Yair Carmon, Aditi Raghunathan, Ludwig Schmidt, Percy Liang, and John C Duchi. Unlabeled data improves adversarial robustness. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Alvin Chan, Yi Tay, Yew Soon Ong, and Jie Fu. Jacobian adversarially regularized networks for robustness. In *International Conference on Learning Representations (ICLR)*, 2020.

Jinghui Chen and Quanquan Gu. Rays: A ray searching method for hard-label adversarial attack. In *International Conference on Knowledge Discovery & Data Mining (KDD)*, 2020.

Jinghui Chen, Yuan Cao, and Quanquan Gu. Benign overfitting in adversarially robust linear classification. *arXiv preprint arXiv:2112.15250*, 2021a.

Tianlong Chen, Zhenyu Zhang, Sijia Liu, Shiyu Chang, and Zhangyang Wang. Robust overfitting may be mitigated by properly learned smoothening. In *International Conference on Learning Representations (ICLR)*, 2021b.

Jeremy M Cohen, Elan Rosenfeld, and J Zico Kolter. Certified adversarial robustness via randomized smoothing. In *International Conference on Machine Learning (ICML)*, 2019.

Keith Conrad. Equivalence of norms. *Expository Paper, University Of Connecticut*, 2018.

Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International Conference on Machine Learning (ICML)*, 2020.

Francesco Croce, Maksym Andriushchenko, Vikash Sehwag, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. Robustbench: a standardized adversarial robustness benchmark. *arXiv preprint arXiv:2010.09670*, 2020.

Imre Csiszár and János Körner. *Information theory: coding theorems for discrete memoryless systems*. Cambridge University Press, 2011.

Gavin Weiguang Ding, Yash Sharma, Kry Yik Chau Lui, and Ruitong Huang. Mma training: Direct input space margin maximization through adversarial training. In *International Conference on Learning Representations (ICLR)*, 2020.

Yinpeng Dong, Qi-An Fu, Xiao Yang, Tianyu Pang, Hang Su, Zihao Xiao, and Jun Zhu. Benchmarking adversarial robustness. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Dominik Maria Endres and Johannes E Schindelin. A new metric for probability distributions. *IEEE Transactions on Information theory*, 49(7):1858–1860, 2003.

Logan Engstrom, Brandon Tran, Dimitris Tsipras, Ludwig Schmidt, and Aleksander Madry. Exploring the landscape of spatial robustness. In *International Conference on Machine Learning (ICML)*, 2019.

Christian Etmann, Sebastian Lunz, Peter Maass, and Carola-Bibiane Schönlieb. On the connection between adversarial robustness and saliency map interpretability. In *International Conference on Machine Learning (ICML)*, 2019.

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*. Springer series in statistics New York, 2001.

Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations (ICLR)*, 2019.

Justin Gilmer, Luke Metz, Fartash Faghri, Samuel S Schoenholz, Maithra Raghu, Martin Wattenberg, and Ian Goodfellow. Adversarial spheres. *arXiv preprint arXiv:1801.02774*, 2018.

Zeinab Golgooni, Mehrdad Saberi, Masih Eskandar, and Mohammad Hossein Rohban. Zerograd: Mitigating and explaining catastrophic overfitting in fgsm adversarial training. *arXiv preprint arXiv:2103.15476*, 2021.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations (ICLR)*, 2015.

Sven Gowal, Chongli Qin, Jonathan Uesato, Timothy Mann, and Pushmeet Kohli. Uncovering the limits of adversarial training against norm-bounded adversarial examples. *arXiv preprint arXiv:2010.03593*, 2020.

Sven Gowal, Sylvestre-Alvise Rebuffi, Olivia Wiles, Florian Stimberg, Dan Andrei Calian, and Timothy A Mann. Improving robustness using generated data. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Will Grathwohl, Kuan-Chieh Wang, Joern-Henrik Jacobsen, David Duvenaud, Mohammad Norouzi, and Kevin Swersky. Your classifier is secretly an energy based model and you should treat it like one. In *International Conference on Learning Representations (ICLR)*, 2020.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

Dan Hendrycks, Kimin Lee, and Mantas Mazeika. Using pre-training can improve model robustness and uncertainty. In *International Conference on Machine Learning (ICML)*, 2019.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Aapo Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research (JMLR)*, 6(Apr):695–709, 2005.

Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Anish Athalye, Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

P Izmailov, AG Wilson, D Podoprikhin, D Vetrov, and T Garipov. Averaging weights leads to wider optima and better generalization. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, 2018.

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

Peilin Kang and Seyed-Mohsen Moosavi-Dezfooli. Understanding catastrophic overfitting in adversarial training. *arXiv preprint arXiv:2105.02942*, 2021.

Harini Kannan, Alexey Kurakin, and Ian Goodfellow. Adversarial logit pairing. *arXiv preprint arXiv:1803.06373*, 2018.

Simran Kaur, Jeremy Cohen, and Zachary C Lipton. Are perceptually-aligned gradients a general property of robust classifiers? *arXiv preprint arXiv:1910.08640*, 2019.

Hoki Kim, Woojin Lee, and Jaewook Lee. Understanding catastrophic overfitting in single-step adversarial training. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2021.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations (ICLR)*, 2014.

Bai Li, Shiqi Wang, Suman Jana, and Lawrence Carin. Towards understanding fast adversarial training. *arXiv preprint arXiv:2006.03089*, 2020a.

Zichao Li, Liyuan Liu, Chengyu Dong, and Jingbo Shang. Overfitting or underfitting? understand robustness drop in adversarial training. *arXiv preprint arXiv:2010.08034*, 2020b.

Chen Liu, Zhichao Huang, Mathieu Salzmann, Tong Zhang, and Sabine Süsstrunk. On the impact of hard adversarial instances on overfitting in adversarial training. *arXiv preprint arXiv:2112.07324*, 2021.

Peter Lorenz, Dominik Strassel, Margret Keuper, and Janis Keuper. Is robustbench/autoattack a suitable benchmark for adversarial robustness? In *The AAAI Workshop on Adversarial Machine Learning and Beyond*, 2022.

Siwei Lyu. Interpretation and generalization of score matching. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, 2009.

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 (ICLR)*, 2018.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2574–2582, 2016.

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

Yurii E Nesterov. A method for solving the convex programming problem with convergence rate o (1/k$\hat{2}$). In *Dokl. akad. nauk Sssr*, volume 269, pages 543–547, 1983.

Tianyu Pang, Chao Du, and Jun Zhu. Max-mahalanobis linear discriminant analysis networks. In *International Conference on Machine Learning (ICML)*, 2018.

Tianyu Pang, Kun Xu, Chongxuan Li, Yang Song, Stefano Ermon, and Jun Zhu. Efficient learning of generative models via finite-difference score matching. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2020a.

Tianyu Pang, Xiao Yang, Yinpeng Dong, Kun Xu, Hang Su, and Jun Zhu. Boosting adversarial training with hypersphere embedding. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020b.

Tianyu Pang, Xiao Yang, Yinpeng Dong, Hang Su, and Jun Zhu. Bag of tricks for adversarial training. In *International Conference on Learning Representations (ICLR)*, 2021.

Maura Pintor, Fabio Roli, Wieland Brendel, and Battista Biggio. Fast minimum-norm adversarial attacks through adaptive norm constraints. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2021.

Rahul Rade and Seyed-Mohsen Moosavi-Dezfooli. Helper-based adversarial training: Reducing excessive margin to achieve a better accuracy vs. robustness trade-off. In *ICML 2021 Workshop on Adversarial Machine Learning*, 2021.

Aditi Raghunathan, Sang Michael Xie, Fanny Yang, John Duchi, and Percy Liang. Understanding and mitigating the tradeoff between robustness and accuracy. In *International Conference on Machine Learning (ICML)*, 2020.

Sylvestre-Alvise Rebuffi, Sven Gowal, Dan A Calian, Florian Stimberg, Olivia Wiles, and Timothy Mann. Fixing data augmentation to improve adversarial robustness. In *Advances in neural information processing systems (NeurIPS)*, 2021.

Leslie Rice, Eric Wong, and J Zico Kolter. Overfitting in adversarially robust deep learning. In *International Conference on Machine Learning (ICML)*, 2020.

Jérôme Rony, Luiz G Hafemann, Luiz S Oliveira, Ismail Ben Ayed, Robert Sabourin, and Eric Granger. Decoupling direction and norm for efficient gradient-based l2 adversarial attacks and defenses. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Andras Rozsa, Manuel Günther, and Terrance E Boult. Are accuracy and robustness correlated. In *IEEE international conference on machine learning and applications (ICMLA)*, pages 227–232. IEEE, 2016.

Shibani Santurkar, Andrew Ilyas, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Image synthesis with a single (robust) classifier. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Amartya Sanyal, Puneet K. Dokania, Varun Kanade, and Philip Torr. How benign is benign overfitting ? In *International Conference on Learning Representations (ICLR)*, 2021.

Ludwig Schmidt, Shibani Santurkar, Dimitris Tsipras, Kunal Talwar, and Aleksander Madry. Adversarially robust generalization requires more data. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 5019–5031, 2018.

Vikash Sehwag, Saeed Mahloujifar, Tinashe Handina, Sihui Dai, Chong Xiang, Mung Chiang, and Prateek Mittal. Improving adversarial robustness using proxy distributions. *arXiv preprint arXiv:2104.09425*, 2021.

Ali Shafahi, Mahyar Najibi, Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein. Adversarial training for free! In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Carl-Johann Simon-Gabriel, Yann Ollivier, Leon Bottou, Bernhard Schölkopf, and David Lopez-Paz. First-order adversarial vulnerability of neural networks and input dimension. In *International Conference on Machine Learning (ICML)*, 2019.

Vasu Singla, Sahil Singla, Soheil Feizi, and David Jacobs. Low curvature activations reduce overfitting in adversarial training. In *IEEE International Conference on Computer Vision (ICCV)*, pages 16423–16433, 2021.

Leslie N Smith and Nicholay Topin. Super-convergence: Very fast training of neural networks using large learning rates. In *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications*, volume 11006, page 1100612. International Society for Optics and Photonics, 2019.

Chuanbiao Song, Kun He, Liwei Wang, and John E Hopcroft. Improving the generalization of adversarial training with domain adaptation. In *International Conference on Learning Representations (ICLR)*, 2019a.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 11895–11907, 2019.

Yang Song, Sahaj Garg, Jiaxin Shi, and Stefano Ermon. Sliced score matching: A scalable approach to density and score estimation. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, 2019b.

Gaurang Sriramanan, Sravanti Addepalli, Arya Baburaj, et al. Guided adversarial attack for evaluating and enhancing adversarial defenses. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

David Stutz, Matthias Hein, and Bernt Schiele. Disentangling adversarial robustness and generalization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

David Stutz, Matthias Hein, and Bernt Schiele. Confidence-calibrated adversarial training: Generalizing to unseen attacks. In *International Conference on Machine Learning (ICML)*, 2020.

Dong Su, Huan Zhang, Hongge Chen, Jinfeng Yi, Pin-Yu Chen, and Yupeng Gao. Is robustness the cost of accuracy? – a comprehensive study on the robustness of 18 deep image classification models. In *The European Conference on Computer Vision (ECCV)*, 2018.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations (ICLR)*, 2014.

Shiyu Tang, Ruihao Gong, Yan Wang, Aishan Liu, Jiakai Wang, Xinyun Chen, Fengwei Yu, Xianglong Liu, Dawn Song, Alan Yuille, et al. Robustart: Benchmarking robustness on architecture design and training techniques. *arXiv preprint arXiv:2109.05211*, 2021.

Guanhong Tao, Shiqing Ma, Yingqi Liu, and Xiangyu Zhang. Attacks meet interpretability: Attribute-steered detection of adversarial samples. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

Florian Tramèr and Dan Boneh. Adversarial training and robustness for multiple perturbations. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 5858–5868, 2019.

Florian Tramèr, Jens Behrmann, Nicholas Carlini, Nicolas Papernot, and Jörn-Henrik Jacobsen. Fundamental tradeoffs between invariance and sensitivity to adversarial perturbations. In *International Conference on Machine Learning (ICML)*. PMLR, 2020.

François Treves. *Topological Vector Spaces, Distributions and Kernels: Pure and Applied Mathematics, Vol. 25*, volume 25. Elsevier, 2016.

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

Yusuke Tsuzuku, Issei Sato, and Masashi Sugiyama. Lipschitz-margin training: Scalable certification of perturbation invariance for deep neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural computation*, 23(7):1661–1674, 2011.

S Vivek B and R Venkatesh Babu. Single-step adversarial training with dropout scheduling. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Abraham Wald. Statistical decision functions which minimize the maximum risk. *Annals of Mathematics*, pages 265–280, 1945.

Haotao Wang, Tianlong Chen, Shupeng Gui, Ting-Kuei Hu, Ji Liu, and Zhangyang Wang. Once-for-all adversarial training: In-situ tradeoff between robustness and accuracy for free. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020a.

Yisen Wang, Difan Zou, Jinfeng Yi, James Bailey, Xingjun Ma, and Quanquan Gu. Improving adversarial robustness requires revisiting misclassified examples. In *International Conference on Learning Representations (ICLR)*, 2020b.

Eric Wong, Leslie Rice, and J. Zico Kolter. Fast is better than free: Revisiting adversarial training. In *International Conference on Learning Representations (ICLR)*, 2020.

Boxi Wu, Jinghui Chen, Deng Cai, Xiaofei He, and Quanquan Gu. Do wider neural networks really help adversarial robustness? In *Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2021.

Dongxian Wu, Shu-Tao Xia, and Yisen Wang. Adversarial weight perturbation helps robust generalization. *Advances in Neural Information Processing Systems (NeurIPS)*, 33, 2020.

Cihang Xie, Mingxing Tan, Boqing Gong, Jiang Wang, Alan Yuille, and Quoc V Le. Adversarial examples improve image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Yao-Yuan Yang, Cyrus Rashtchian, Hongyang Zhang, Ruslan Salakhutdinov, and Kamalika Chaudhuri. A closer look at accuracy vs. robustness. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Yaodong Yu, Zitong Yang, Edgar Dobriban, Jacob Steinhardt, and Yi Ma. Understanding generalization in adversarial training via the bias-variance decomposition. *arXiv preprint arXiv:2103.09947*, 2021.

Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *The British Machine Vision Conference (BMVC)*, 2016.

Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P Xing, Laurent El Ghaoui, and Michael I Jordan. Theoretically principled trade-off between robustness and accuracy. In *International Conference on Machine Learning (ICML)*, 2019.

Jingfeng Zhang, Xilie Xu, Bo Han, Gang Niu, Lizhen Cui, Masashi Sugiyama, and Mohan Kankanhalli. Attacks which do not kill training make adversarial learning stronger. In *International Conference on Machine Learning (ICML)*, 2020.

Jingfeng Zhang, Jianing Zhu, Gang Niu, Bo Han, Masashi Sugiyama, and Mohan Kankanhalli. Geometry-aware instance-reweighted adversarial training. In *International Conference on Learning Representations (ICLR)*, 2021a.

Yihua Zhang, Guanhuan Zhang, Prashant Khanduri, Mingyi Hong, Shiyu Chang, and Sijia Liu. Revisiting and advancing fast adversarial training through the lens of bi-level optimization. *arXiv preprint arXiv:2112.12376*, 2021b.

A. Proofs

In this section, we provide proofs of the Theorems proposed in the main text.

A.1. Proof of Theorem 1

According to the symmetry and the triangle inequality of any distance metric $\mathcal{D}(\cdot||\cdot)$, we have

$$\begin{aligned}
&\mathbf{R}_{\text{SCORE}}^{\mathcal{D}}(\theta)\\
=&\mathbb{E}_{p_d(x)}\left[\max_{x'\in B(x)}\mathcal{D}\left(p_d(y|x')\middle\|p_\theta(y|x')\right)\right]\\
\leq&\mathbb{E}_{p_d(x)}\left[\max_{x'\in B(x)}\left(\mathcal{D}\left(p_d(y|x')\middle\|p_d(y|x)\right)+\mathcal{D}\left(p_d(y|x)\middle\|p_\theta(y|x')\right)\right)\right]\\
\leq&\mathbb{E}_{p_d(x)}\left[\max_{x'\in B(x)}\mathcal{D}\left(p_d(y|x')\middle\|p_d(y|x)\right)+\max_{x'\in B(x)}\mathcal{D}\left(p_d(y|x)\middle\|p_\theta(y|x')\right)\right]\\
=&\mathbb{E}_{p_d(x)}\left[\max_{x'\in B(x)}\mathcal{D}\left(p_d(y|x)\middle\|p_\theta(y|x')\right)\right]+C^{\mathcal{D}}\\
=&\mathbf{R}_{\text{Madry}}^{\mathcal{D}}(\theta)+C^{\mathcal{D}}.
\end{aligned}\tag{14}$$

Furthermore, we can show that

$$\begin{aligned}
&\mathbf{R}_{\text{SCORE}}^{\mathcal{D}}(\theta)+C^{\mathcal{D}}\\
=&\mathbb{E}_{p_d(x)}\left[\max_{x'\in B(x)}\mathcal{D}\left(p_d(y|x')\middle\|p_\theta(y|x')\right)+\max_{x'\in B(x)}\mathcal{D}\left(p_d(y|x')\middle\|p_d(y|x)\right)\right]\\
\geq&\mathbb{E}_{p_d(x)}\left[\max_{x'\in B(x)}\left(\mathcal{D}\left(p_d(y|x')\middle\|p_d(y|x)\right)+\mathcal{D}\left(p_d(y|x')\middle\|p_\theta(y|x')\right)\right)\right]\\
\geq&\mathbb{E}_{p_d(x)}\left[\max_{x'\in B(x)}\mathcal{D}\left(p_d(y|x)\middle\|p_\theta(y|x')\right)\right]\\
=&\mathbf{R}_{\text{Madry}}^{\mathcal{D}}(\theta).
\end{aligned}\tag{15}$$

Similarly, there is $\mathbf{R}_{\text{Madry}}^{\mathcal{D}}(\theta)+\mathbf{R}_{\text{SCORE}}^{\mathcal{D}}(\theta)\geq C^{\mathcal{D}}$, thus in conclusion, we prove that

$$|\mathbf{R}_{\text{Madry}}^{\mathcal{D}}(\theta)-C^{\mathcal{D}}|\leq\mathbf{R}_{\text{SCORE}}^{\mathcal{D}}(\theta)\leq\mathbf{R}_{\text{Madry}}^{\mathcal{D}}(\theta)+C^{\mathcal{D}}.\tag{16}$$

□

A.2. Proof of Theorem 2

According to Theorem 1, it is easy to show that

$$|\mathbf{R}_{\text{SCORE}}^{\mathcal{D}}(\theta)-C^{\mathcal{D}}|\leq\mathbf{R}_{\text{Madry}}^{\mathcal{D}}(\theta),\tag{17}$$

Since $\phi(\cdot)$ is monotonically increasing, there is

$$\arg\max_{x'\in B(x)}\mathcal{D}\left(p_d(y|x)\middle\|p_\theta(y|x')\right)=\arg\max_{x'\in B(x)}\phi\circ\mathcal{D}\left(p_d(y|x)\middle\|p_\theta(y|x')\right).\tag{18}$$

Additionally $\phi(\cdot)$ is convex, then by Jensen's inequality, we have

$$\phi\left(\mathbf{R}_{\text{Madry}}^{\mathcal{D}}(\theta)\right)\leq\mathbf{R}_{\text{Madry}}^{\phi\circ\mathcal{D}}(\theta).\tag{19}$$

Take this formula into Eq. (17), we prove that

$$|\mathbf{R}_{\text{SCORE}}^{\mathcal{D}}(\theta)-C^{\mathcal{D}}|\leq\phi^{-1}\left(\mathbf{R}_{\text{Madry}}^{\phi\circ\mathcal{D}}(\theta)\right).\tag{20}$$

□

A.3. Proof of Theorem 3

As to the TRADES objective, we have

$$\begin{aligned}
&\mathbf{R}^{\mathcal{D}}_{\text{TRADES}}(\theta;\beta)\\
=&\mathbb{E}_{p_d(x)}\left[\mathcal{D}\left(p_d(y|x)\middle\|p_\theta(y|x)\right)+\beta\cdot\max_{x'\in B(x)}\mathcal{D}\left(p_\theta(y|x)\middle\|p_\theta(y|x')\right)\right]\\
\leq&\mathbb{E}_{p_d(x)}\left[\mathcal{D}\left(p_d(y|x)\middle\|p_\theta(y|x)\right)+\beta\cdot\max_{x'\in B(x)}\left(\mathcal{D}\left(p_d(y|x)\middle\|p_\theta(y|x)\right)+\mathcal{D}\left(p_d(y|x)\middle\|p_\theta(y|x')\right)\right)\right]\\
=&\mathbb{E}_{p_d(x)}\left[(1+\beta)\cdot\mathcal{D}\left(p_d(y|x)\middle\|p_\theta(y|x)\right)+\beta\cdot\max_{x'\in B(x)}\mathcal{D}\left(p_d(y|x)\middle\|p_\theta(y|x')\right)\right]\\
\leq&\mathbb{E}_{p_d(x)}\left[(1+2\beta)\cdot\max_{x'\in B(x)}\mathcal{D}\left(p_d(y|x)\middle\|p_\theta(y|x')\right)\right]\\
=&(1+2\beta)\cdot\mathbf{R}^{\mathcal{D}}_{\text{Madry}}(\theta)
\end{aligned}\tag{21}$$

Besides, it is easy to show that $\mathbf{R}^{\mathcal{D}}_{\text{TRADES}}(\theta;\beta)\geq\mathbf{R}^{\mathcal{D}}_{\text{Madry}}(\theta)$ holds for any $\beta\geq 1$. □

A.4. Proof of Theorem 4

In the ℓ_p-norm threat model of $B(x)=\{x'\big|\|x'-x\|_p\leq\epsilon\}$, we take the first-order expansion similar as in Simon-Gabriel et al. (2019), there is

$$\mathbf{R}^{\mathcal{D}}_{\text{SCORE}}(\theta)-\mathbf{R}^{\mathcal{D}}_{\text{Standard}}(\theta)=\epsilon\cdot\mathbb{E}_{p_d(x)}\left[\left\|\nabla_x\mathcal{D}\left(p_d(y|x)\middle\|p_\theta(y|x)\right)\right\|_q\right]+o(\epsilon).\tag{22}$$

In particular, when $\mathcal{D}$ is the ℓ_1-distance metric, we have

$$\mathbf{R}^{\ell_1}_{\text{SCORE}}(\theta)-\mathbf{R}^{\ell_1}_{\text{Standard}}(\theta)=\epsilon\cdot\mathbb{E}_{p_d(x)}\left[\left\|\sum_y s_y\cdot\left(\nabla_x p_d(y|x)-\nabla_x p_\theta(y|x)\right)\right\|_q\right]+o(\epsilon),\tag{23}$$

where $s_y=\text{sign}(p_d(y|x)-p_\theta(y|x))$. Under the assumption, there is $s_y=1$ for $y=\mathcal{Y}_d(x)$; otherwise $s_y=-1$. Besides, since $\sum_y p_d(y|x)=\sum_y p_\theta(y|x)=1$, we know that

$$\sum_y\nabla_x p_d(y|x)=\sum_y\nabla_x p_\theta(y|x)=0.\tag{24}$$

Take these into Eq. (23), we can derive that

$$\mathbf{R}^{\ell_1}_{\text{SCORE}}(\theta)-\mathbf{R}^{\ell_1}_{\text{Standard}}(\theta)=2\epsilon\cdot\mathbb{E}_{p_d(x)}\left[\|\nabla_x p_d(\mathcal{Y}_d(x)|x)-\nabla_x p_\theta(\mathcal{Y}_d(x)|x)\|_q\right]+o(\epsilon).\tag{25}$$

□

A.5. Proof of Theorem 5

Under Gaussian augmentation, the augmented joint distribution of (x,y) becomes

$$\begin{aligned}
p_d^\sigma(x,y)&=\int\mathcal{N}(\omega;0,I)\cdot p_d(x-\sqrt{\sigma}\omega,y)d\omega\\
&=\mathbb{E}_{\mathcal{N}(\omega;0,I)}\left[p_d(x-\sqrt{\sigma}\omega|y)\right]p_d(y)\\
&=p_d^\sigma(x|y)p_d(y)\\
&\neq p_d^\sigma(x)p_d(y|x).
\end{aligned}\tag{26}$$

Then we can derive the derivatives of augmented cross-entropy loss w.r.t. σ as

$$\begin{aligned}
&\frac{d}{d\sigma}\mathbb{E}_{p_d(y)}\mathbb{E}_{p_d^\sigma(x|y)}\left[-\log p_\theta(y|x)\right]\\
=&-\mathbb{E}_{p_d(y)}\int\frac{d}{d\sigma}p_d^\sigma(x|y)\log p_\theta(y|x)dx\\
=&-\frac{1}{2}\mathbb{E}_{p_d(y)}\int\nabla_x^\top\nabla_x p_d^\sigma(x|y)\log p_\theta(y|x)dx && \text{(diffusion equation)}\\
=&-\frac{1}{2}\mathbb{E}_{p_d(y)}\left[\sum_i\frac{\partial}{\partial x_i}p_d^\sigma(x|y)\log p_\theta(y|x)\Big|_{-\infty}^{\infty}-\int\nabla_x\log p_\theta(y|x)^\top\nabla_x p_d^\sigma(x|y)dx\right] && \text{(integration by parts)}\\
=&\frac{1}{2}\mathbb{E}_{p_d(y)}\mathbb{E}_{p_d^\sigma(x|y)}\left[\nabla_x\log p_\theta(y|x)^\top\nabla_x\log p_d^\sigma(x|y)\right].
\end{aligned}\tag{27}$$

The term $\sum_i\frac{\partial}{\partial x_i}p_d^\sigma(x|y)\log p_\theta(y|x)\Big|_{-\infty}^{\infty}$ equals to zero under the boundary assumption (Hyvärinen, 2005). □

A.6. Proof of Corollary 1

Let $\phi(\cdot)=(\cdot)^2$ be the square function, which is monotonically increasing and convex. Take $\phi(\cdot)^{-1}=\sqrt{(\cdot)}$ and $\mathcal{D}$ be ℓ_1-distance into Theorem 2, we have

$$|\mathbf{R}_{\text{SCORE}}^{\ell_1}(\theta)-C^{\ell_1}|\leq\sqrt{\mathbf{R}_{\text{Madry}}^{\ell_1^2}(\theta)}.\tag{28}$$

From the Pinsker's inequality we have

$$\max_{x'\in B(x)}\|p_d(y|x)-p_\theta(y|x')\|_1^2\leq 2\cdot\max_{x'\in B(x)}\text{KL}(p_d(y|x)||p_\theta(y|x')).\tag{29}$$

After taking expectation w.r.t. $p_d(x)$, we achieve that $\mathbf{R}_{\text{Madry}}^{\ell_1^2}(\theta)\leq 2\cdot\mathbf{R}_{\text{Madry}}(\theta)$, which finally proves that

$$|\mathbf{R}_{\text{SCORE}}^{\ell_1}(\theta)-C^{\ell_1}|\leq\sqrt{2\cdot\mathbf{R}_{\text{Madry}}(\theta)}.\tag{30}$$

□

B. Detailed Derivations

In this section, we provide detailed derivations to better support our conclusions.

B.1. The Connection between $\mathbf{R}_{\text{Madry}}(\theta)$ and the Original Objective in Madry et al. (2018)

In Madry et al. (2018), the robust error $\mathbf{R}_{\text{Madry}}(\theta)$ is defined w.r.t. the a loss function $\mathcal{L}(x,y;\theta)$, e.g., the cross-entropy loss, formulated as:

$$\widetilde{\mathbf{R}}_{\text{Madry}}(\theta)=\mathbb{E}_{p_d(x,y)}\left[\max_{x'\in B(x)}\mathcal{L}(x',y;\theta)\right]=\mathbb{E}_{p_d(x)}\left[\sum_y p_d(y|x)\max_{x'\in B(x)}\mathcal{L}(x',y;\theta)\right],\tag{31}$$

where the maximization w.r.t. x' takes place *before* the summation w.r.t. y. By setting $\mathcal{L}(x,y;\theta)=-\log p_\theta(y|x)$ be the cross-entropy loss, we can rewrite our defined $\mathbf{R}_{\text{Madry}}$ as

$$\begin{aligned}
\mathbf{R}_{\text{Madry}}(\theta)=&\mathbb{E}_{p_d(x)}\left[\max_{x'\in B(x)}\text{KL}\left(p_d(y|x)\big\|p_\theta(y|x')\right)\right]\\
=&\mathbb{E}_{p_d(x)}\left[\max_{x'\in B(x)}\sum_y p_d(y|x)\mathcal{L}(x',y;\theta)\right]+\underbrace{\mathbb{E}_{p_d(x)}\left[\sum_y p_d(y|x)\log p_d(y|x)\right]}_{\textbf{independent of }\theta},
\end{aligned}\tag{32}$$

where only the first term works when optimized w.r.t. θ. As seen, in this formula the maximization w.r.t. x' takes place *after* the summation w.r.t. y. Although the summation operator $\sum_y$ and the maximization operator $\max_{x'\in B(x)}$ are not directly permutable, we can permute summation and gradient operators when we use first-order optimization methods, e.g, SGD or Adam. So the (first-order) updating directions of minimizing $\widetilde{\mathbf{R}}_{\text{Madry}}(\theta)$ and $\mathbf{R}_{\text{Madry}}(\theta)$ can be regarded as equal, and we treat them as the same objective in the main text.

B.2. Using Score-based Learning to Optimize $\mathbf{R}_{\text{SCORE}}(\theta)$

To directly minimize SCORE with first-order optimizers, we need to explicitly compute $\nabla_x \text{KL}\left(p_d(y|x) \middle\| p_\theta(y|x)\right)$ as

$$
\begin{aligned}
&\nabla_x \text{KL}\left(p_d(y|x) \middle\| p_\theta(y|x)\right)\\
=&\sum_{y\in[L]} \left[-p_d(y|x)\nabla_x \log p_\theta(y|x) + \nabla_x p_d(y|x)\left(1+\log p_d(y|x) - \log p_\theta(y|x)\right)\right]\\
=&\sum_{y\in[L]} \left[-p_d(y|x)\nabla_x \log p_\theta(y|x) + \nabla_x p_d(y|x)\left(\log p_d(y|x) - \log p_\theta(y|x)\right)\right]\\
=&\sum_{y\in[L]} p_d(y|x)\Big[-\underbrace{\nabla_x \log p_\theta(y|x)}_{\textbf{model gradient}} + \underbrace{\nabla_x \log p_d(y|x)}_{\textbf{data gradient}}\left(\log p_d(y|x) - \log p_\theta(y|x)\right)\Big].
\end{aligned} \tag{33}
$$

As seen, we need to have access to the data gradient $\nabla_x \log p_d(y|x)$. To this end, score matching methods may be applied (Vincent, 2011; Song et al., 2019b; Pang et al., 2020a), based on the decomposition that

$$
\nabla_x \log p_d(y|x) = \nabla_x \log p_d(x|y) - \nabla_x \log p_d(x), \tag{34}
$$

which means that we can estimate a conditional data score $\nabla_x \log p_d(x|y)$ and an unconditional data score $\nabla_x \log p_d(x)$ to obtain $\nabla_x \log p_d(y|x)$. We can use either an energy-based model (EBM) with back-propagation or a direct scorenet (Song and Ermon, 2019) as the data-score model. In our initial experiments, we employ NCSN++ (Song et al., 2019a) to estimate the data scores, using denoising score matching (DSM) (Vincent, 2011). Even though MCMC methods like SGLD can generate high-quality images with the learned scores, we find that the learned scores are of high variance when applied into the discriminative learning process. Therefore, we do not further explore this pipeline in this paper, but we still regard score-based learning as a promising and principled way to optimize SCORE (just as the abbreviation implies).

C. Detailed Discussion on the Effects of Randomized Smoothing

Recall that the Gaussian-augmented cross-entropy loss is defined as

$$
\mathbf{R}_G(\theta;\sigma) = \mathbb{E}_{p_d^\sigma(x,y)}\left[-\log p_\theta(y|x)\right], \tag{35}
$$

where $\mathbf{R}_G(\theta;0)$ is the cross-entropy loss. Then beyond the conclusion in Theorem 5, we can tune the effect of the gradient alignment term in $\mathbf{R}_G(\theta;\sigma)$ by subtracting a scaled $\mathbf{R}_G(\theta;0)$, controlled via an extra hyperparameter γ as

$$
\begin{aligned}
&\frac{1}{1-\gamma}\cdot\left[\mathbf{R}_G(\theta;\sigma) - \gamma\cdot\mathbf{R}_G(\theta;0)\right]\\
=&\mathbf{R}_G(\theta;0) + \frac{\sigma}{2(1-\gamma)}\cdot\mathbb{E}_{p_d^\sigma(x,y)}\left[\nabla_x \log p_\theta(y|x)^\top \nabla_x \log p_d^\sigma(x|y)\right] + o(\sigma).
\end{aligned} \tag{36}
$$

Note that the coefficient of the gradient alignment term is now $\frac{\sigma}{2(1-\gamma)}$. Thus, we can magnify the coefficient by increasing γ close to one, rather than increasing σ which could degrade clean performance. Besides, we can adaptively anneal γ and σ during training to keep $\frac{\sigma}{2(1-\gamma)}$ as a constant, while the saliency map of the trained models is more interpretable. In the implementation, σ can be uniformly sampled for each data point, and γ could point-wisely adapt to the value of σ. Similarly, we can also construct

$$
\begin{aligned}
&\frac{1}{\gamma-1}\cdot\left[\gamma\cdot\mathbf{R}_G(\theta;0) - \mathbf{R}_G(\theta;\sigma)\right]\\
=&\mathbf{R}_G(\theta;0) - \frac{\sigma}{2(\gamma-1)}\cdot\mathbb{E}_{p_d^\sigma(x,y)}\left[\nabla_x \log p_\theta(y|x)^\top \nabla_x \log p_d^\sigma(x|y)\right] + o(\sigma).
\end{aligned} \tag{37}
$$

Interestingly, by comparing the gradient alignment terms in Eq. (36) and Eq. (37), the sign is reversed, i.e., we can even control the direction of gradient alignment.

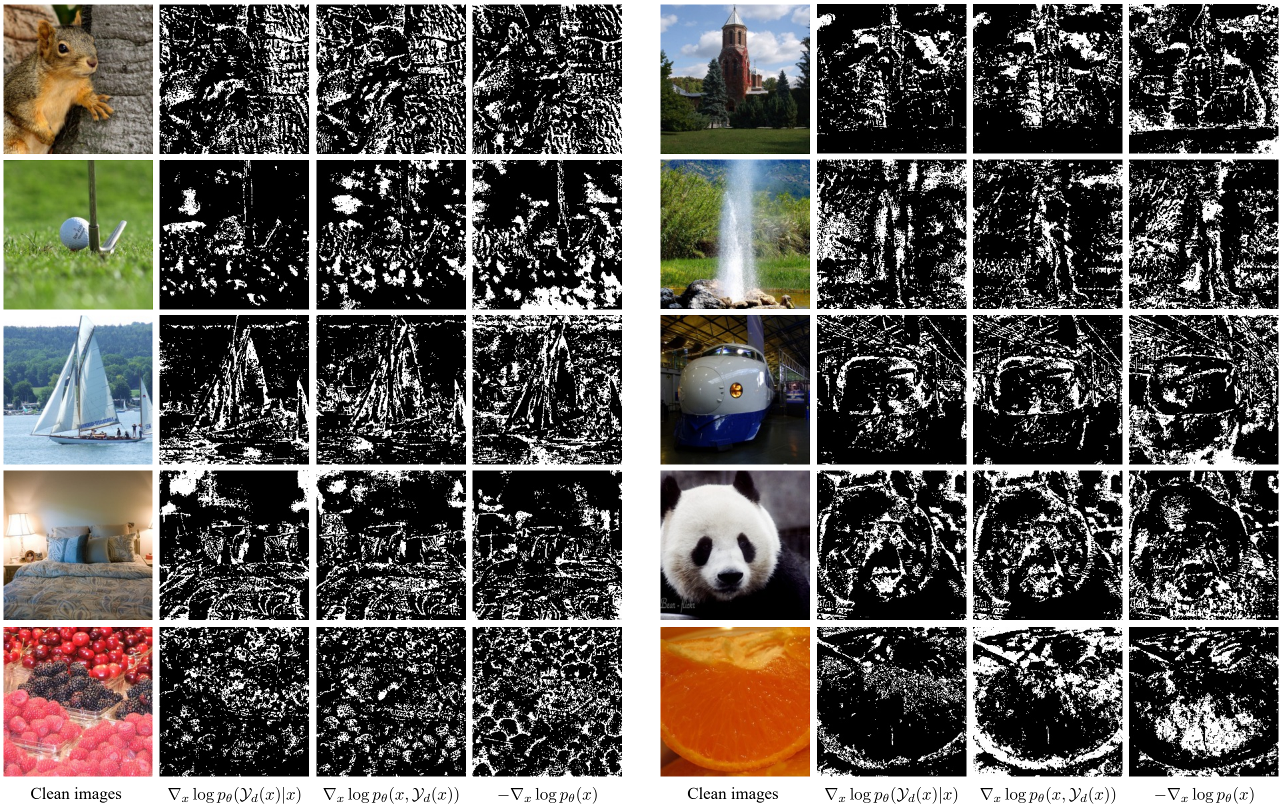
Clean images
$\nabla_x \log p_\theta(\mathcal{Y}_d(x)|x)$
$\nabla_x \log p_\theta(x, \mathcal{Y}_d(x))$
$-\nabla_x \log p_\theta(x)$
Clean images
$\nabla_x \log p_\theta(\mathcal{Y}_d(x)|x)$
$\nabla_x \log p_\theta(x, \mathcal{Y}_d(x))$
$-\nabla_x \log p_\theta(x)$

Figure 5. More examples of visualizing the semantic gradients of a ResNet-50 trained by FreeAT on ImageNet, using the same pipeline as in Fig. 4. The gradients will be noisy and almost uniformly scattered for standardly trained models (not plotted here).

The gradient alignment cannot be perfectly achieved. Since $\sum_y p_\theta(y|x) = 1$, there is $\sum_y \nabla_x p_\theta(y|x) = 0$. On the other hand, there is $\sum_y p_d(x|y)p_d(y) = p_d(x)$ and thus $\sum_y p_d(y)\nabla_x p_d(x|y) = \nabla_x p_d(x)$, which indicates that $p_\theta(y|x)$ cannot perfectly align with $p_d(x|y)$ or $-p_d(x|y)$ especially when $p_d(y)$ is an uniform distribution.

Why does the gradient alignment come from training rather than inference? Except for the Gaussian-augmented training, the most critical characteristic of randomized smoothing is to apply an ensemble of Gaussian-perturbed predictions during inference, formulated as

$$p_\theta^\sigma(y|x) = \mathbb{E}_{\mathcal{N}(\omega;0,I)}\left[p_\theta(y|x + \sqrt{\sigma}\omega)\right]. \tag{38}$$

In practice, the expectation is approximated by N samples of $\omega_1, \cdots, \omega_N$. So it is probable that the inference mechanism causes the semantic gradients. However, we notice that in Fig. 14 of Kaur et al. (2019), the authors ablate on the number of Monte Carlo samples used for gradient estimation. As observed, even for $N = 1$ (i.e., using a single Gaussian-perturbed prediction), the gradients look perceptually aligned, which is certainly not the case for a standardly trained model (i.e., without Gaussian augmentation). Thus we attribute the semantic phenomenon more to the training phase.

D. Additional Experiments

In this section, we provide more technical details and empirical results.

D.1. Visualization of (KL-based) Overfitting

Recall the discussion in Sec. 4.1, actually when $\mathbf{R}_{\text{SCORE}}^{\ell_1}(\theta) = 0$, there is $p_\theta(y|x) = p_d(y|x)$. Thus, in this case we have $\mathbf{R}_{\text{Madry}}(\theta) = C^{\text{KL}}$, where

$$C^{\text{KL}} = \mathbb{E}_{p_d(x)}\left[\max_{x' \in B(x)} \text{KL}\left(p_d(y|x) \| p_d(y|x')\right)\right]. \tag{39}$$

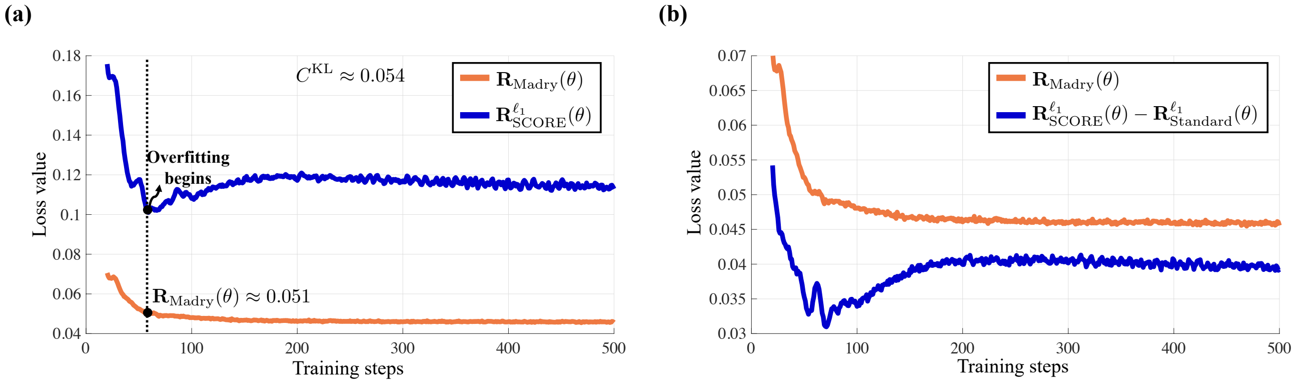

Figure 6. Toy demo with the same settings as in Fig. 1. **(a)** Illustration of Corollary 1 on the overfitting phenomenon. $\mathbf{R}^{\ell_1}_{\mathrm{SCORE}}(\theta)$ begins to overfit when $\mathbf{R}_{\mathrm{Madry}}(\theta)$ is minimized to around C^{KL}; **(b)** Illustration of Theorem 4 on the phenomenon of semantic gradients. When minimizing $\mathbf{R}_{\mathrm{Madry}}(\theta)$, the gradient alignment term, i.e., $\mathbf{R}^{\ell_1}_{\mathrm{SCORE}}(\theta) - \mathbf{R}^{\ell_1}_{\mathrm{Standard}}(\theta) \approx 2\epsilon \cdot \mathbb{E}_{p_d(x)}\left[\left\|\nabla_x p_d(\mathcal{Y}_d(x)|x) - \nabla_x p_\theta(\mathcal{Y}_d(x)|x)\right\|_q\right]$ keeps decreasing before $\mathbf{R}^{\ell_1}_{\mathrm{SCORE}}(\theta)$ is minimized to be less than C^{ℓ_1}.

According to Pinsker's inequality, there is $C^{\mathrm{KL}} \geq (C^{\ell_1})^2/2$. This implies that we should early-stop $\mathbf{R}_{\mathrm{Madry}}(\theta)$ even earlier at C^{KL}. In Fig. 6 (a), we minimize $\mathbf{R}_{\mathrm{Madry}}(\theta)$ in training, and find that $\mathbf{R}^{\ell_1}_{\mathrm{SCORE}}(\theta)$ indeed begins to overfit when $\mathbf{R}_{\mathrm{Madry}}(\theta) \approx C^{\mathrm{KL}}$. Nevertheless, Corollary 1 analytically describes how minimizing $\mathbf{R}_{\mathrm{Madry}}(\theta)$ affects $\mathbf{R}^{\ell_1}_{\mathrm{SCORE}}(\theta)$.

D.2. Visualization of Semantic Gradients

In Fig. 4 and Fig. 5, we visualize the semantic input gradients of a ResNet-50 trained by FreeAT on ImageNet. Unlike previous work that directly plots all the perturbation after normalization (Pang et al., 2020b), we add up the partial derivatives of three RGB channels for each pixel position, and sort out the top 10% pixel positions with large values of total derivatives (i.e., affect the objectives the most) in the plots. This way can highlight the shape-based characteristics learned by the robust models. Since $\log p_\theta(\mathcal{Y}_d(x)|x)$ is a discriminative objective, we also consider $\log p_\theta(x, \mathcal{Y}_d(x))$ and $\log p_\theta(x)$ based on the fact that $\log p_\theta(\mathcal{Y}_d(x)|x) = \log p_\theta(x, \mathcal{Y}_d(x)) - \log p_\theta(x)$, and similarly $\nabla_x \log p_\theta(\mathcal{Y}_d(x)|x) = \nabla_x \log p_\theta(x, \mathcal{Y}_d(x)) - \nabla_x \log p_\theta(x)$. According to Grathwohl et al. (2020), given a classifier $p_\theta(y|x)$ with a softmax final layer, i.e.,

$$p_\theta(y|x) = \frac{\exp(f_\theta(x)[y])}{\sum_{y'} \exp(f_\theta(x)[y'])}, \tag{40}$$

where $f_\theta(x)[y]$ indicates the y^{th} index of $f_\theta(x)$, i.e., the logit corresponding the the class label y. Then we can reinterpret the discriminative model as an EBM for the joint distribution $p_\theta(x, y)$, formulated as

$$p_\theta(x, y) = \frac{\exp(f_\theta(x)[y])}{Z(\theta)}; p_\theta(x) = \frac{\sum_{y'} \exp(f_\theta(x)[y'])}{Z(\theta)}, \tag{41}$$

where $Z(\theta)$ is the unknown normalizing constant. Since there is $\nabla_x \log Z(\theta) = 0$, we have $\nabla_x \log p_\theta(x, y) = \nabla_x f_\theta(x)[y]$ and $\nabla_x \log p_\theta(x) = \nabla_x \log \sum_{y'} \exp(f_\theta(x)[y'])$. We can compute $\nabla_x \log p_\theta(x)$ using the numerically stable operator of Log-Sum-Exp. Note that in the figures, we plot $\nabla_x \log p_\theta(x, \mathcal{Y}_d(x))$ (i.e., the fastest direction of *increasing* $p_\theta(x, \mathcal{Y}_d(x))$) and $-\nabla_x \log p_\theta(x)$ (i.e., the fastest direction of *decreasing* $p_\theta(x)$).

D.3. Checking for Gradient Obfuscation

Following the evaluation guide by Carlini et al. (2019), we perform sanity check for gradient obfuscation (Athalye et al., 2018). In Fig. 7, we apply PGD-40 attacks with different values of perturbation sizes. We apply five restarts since more restarts only marginally lower the robust accuracy. As seen, on both CIFAR-10 and CIFAR-100, the model accuracy finally converges to zero, indicating that the SE function does not lead to gradient obfuscation.

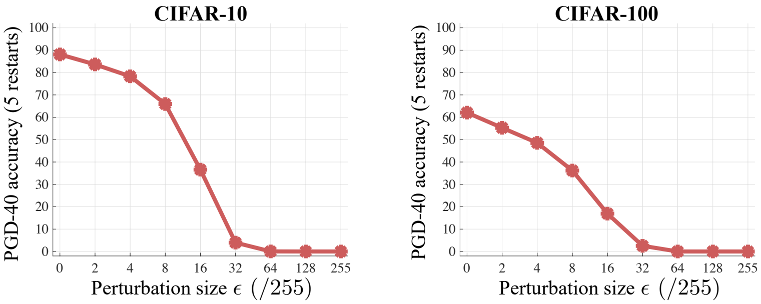

Figure 7. Sanity check on gradient obfuscation. We employ ℓ_∞ threat model and PGD-40 attacks, with perturbation size ϵ increasing from $0/255$ to $255/255$. We evaluate the WRN-28-10 models trained by our methods (the SE function) on CIFAR-10 and CIFAR-100.

E. Clarification on Backgrounds

This section clarifies some concepts discussed in our papers to avoid ambiguity.

E.1. Overfitting; Catastrophic Overfitting; Benign Overfitting

In Sec. 4.1, the *overfitting* phenomenon refers to the one observed in multi-step adversarial training (Rice et al., 2020). In contrast, there is another analogous concept, named *catastrophic overfitting*, which is observed in one-step adversarial training (Wong et al., 2020). While we focus on the former overfitting in the sense of a generalization gap, much work focuses on the latter one to alleviate or understand catastrophic overfitting in which the robust accuracy rapidly drops (Andriushchenko and Flammarion, 2020; Li et al., 2020a; Vivek B and Venkatesh Babu, 2020; Kang and Moosavi-Dezfooli, 2021; Zhang et al., 2021b; Kim et al., 2021; Golgooni et al., 2021). Recently, the concept of *benign overfitting* has also drawn attention (Sanyal et al., 2021; Chen et al., 2021a), studying the phenomenon that classifiers memorize noisy training data yet still achieve a good generalization performance.

E.2. Other Trade-offs in the Adversarial Literature

This paper focuses on the robustness-accuracy trade-off, which refers to the observations that a model achieves a higher robust accuracy (usually via adversarial training) at the cost of degraded clean accuracy. Nevertheless, several other trade-off phenomena are studied in previous work. For examples, Engstrom et al. (2019) show that state-of-the-art ℓ_∞ robust models turn out to be vulnerable to translations and rotations; Tramèr and Boneh (2019) demonstrate a trade-off in robustness to different types of ℓ_p-bounded (e.g., ℓ_1, ℓ_2, and ℓ_∞) perturbations; Tramèr et al. (2020) present fundamental trade-offs between sensitivity-based and invariance-based adversarial examples.

E.3. AutoAttack for Evaluation

In our experiments, we mainly apply AutoAttack for evaluating the robustness of baselines and our methods. We have to clarify that there are many other benchmarks and relevant attacks in the literature (Chen and Gu, 2020; Dong et al., 2020; Sriramanan et al., 2020; Tang et al., 2021), while recent work also argues that the adversarial examples crafted by AutoAttack are easy to be detected (Lorenz et al., 2022). Nevertheless, RobustBench[4] (based on AutoAttack) is widely recognized and a challenging benchmark, with frequent updates on the state-of-the-art models. Achieving top-rank performance on RobustBench is a compelling evidence for the effectiveness of the proposed method.

[4] https://robustbench.github.io/