DRF: Improving Certified Robustness via Distributional Robustness Framework

Zekai Wang, Zhengyu Zhou, Weiwei Liu*

School of Computer Science,
Institute of Artificial Intelligence,
National Engineering Research Center for Multimedia Software,
Hubei Key Laboratory of Multimedia and Network Communication Engineering,
Wuhan University, China
{wzekai99, zzysince1999, liuweiwei863}@gmail.com

Abstract

Randomized smoothing (RS) has provided state-of-the-art (SOTA) certified robustness against adversarial perturbations for large neural networks. Among studies in this field, methods based on adversarial training (AT) achieve remarkably robust performance by applying adversarial examples to construct the smoothed classifier. These AT-based RS methods typically seek a pointwise adversary that generates the worst-case adversarial examples by perturbing each input independently. However, there are unexplored benefits to considering such adversarial robustness across the entire data distribution. To this end, we provide a novel framework called DRF, which connects AT-based RS methods with distributional robustness (DR), and show that these methods are special cases of their counterparts in our framework. Due to the advantages conferred by DR, our framework can control the trade-off between the clean accuracy and certified robustness of smoothed classifiers to a significant extent. Our experiments demonstrate that DRF can substantially improve the certified robustness of AT-based RS.

1 Introduction

While neural networks (NNs) have achieved remarkable performance in various applications (He et al. 2016; Devlin et al. 2019; Silver et al. 2017; Mao et al. 2021, 2022, 2023), many empirical and theoretical studies (Xu and Liu 2022, 2023; Chen and Liu 2023) have demonstrated that NNs are vulnerable when dealing with imperceptibly perturbed images, referred to as adversarial examples. Adversarial attacks (Goodfellow, Shlens, and Szegedy 2015; Madry et al. 2018; Carlini and Wagner 2017) are usually generated by adding small perturbations to benign images; notably, these minor perturbations can drastically change the predictions of an NN-based classifier, even when the perturbation has no effect whatsoever on the semantic information perceived by humans.

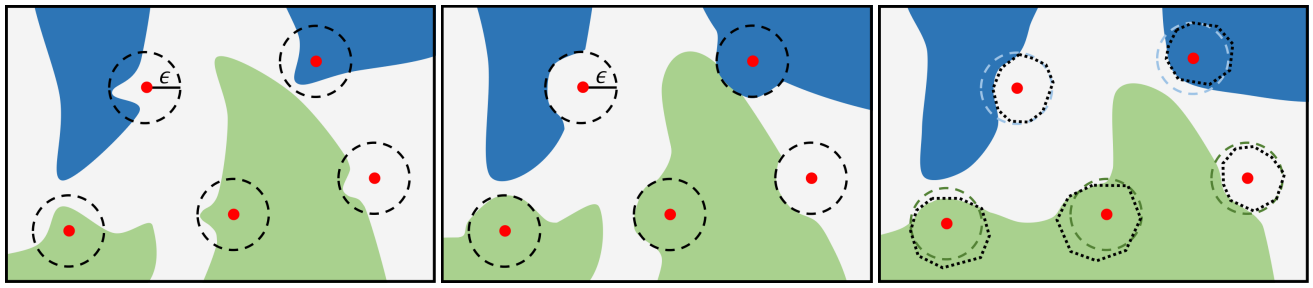
This intriguing weakness of NN has motivated a rapidly growing body of work focused on obtaining a robust NN model (Li, Zou, and Liu 2022; Zou and Liu 2023a,b; Shi and Liu 2023). Unfortunately, most defense heuristics have subsequently been shown to fail against suitably powerful attack algorithms (Carlini and Wagner 2017; Uesato et al. 2018; Tramèr et al. 2020). Even if the model is made robust to the attack algorithm used for evaluation, there is no guarantee that it will remain robust to other unseen attacks. This has encouraged researchers to develop *certified robustness* (Katz et al. 2017; Wong et al. 2018; Wang and Liu 2022, 2023): *i.e.*, regardless of what attack algorithm is applied, classifiers whose prediction at point x is certified to be constant within a neighborhood of x.

Randomized smoothing (RS) (Lécuyer et al. 2019; Cohen, Rosenfeld, and Kolter 2019) is a promising method that can provide certified robustness for large NNs. Cohen, Rosenfeld, and Kolter (2019) show that any classifier can be transformed into a certified robust classifier by averaging its predictions over Gaussian noise, with the certified robustness depending on how well the classifier performs when faced with the noise. At present, RS is considered the SOTA approach to offering a provable guarantee of robustness against L_2-perturbations (Li et al. 2020). In light of this, many existing works focus on improving the robustness guarantee given by RS, such as by using different smoothing measures (Lee et al. 2019; Yang et al. 2020), different divergences (Dvijotham et al. 2020), etc.

One important direction in this line of research is that of investigating which training of the base classifier can maximize the certified robustness after smoothing. Empirically, RS methods employing adversarial training can outperform Gaussian-based RS and significantly improve the certified robustness. Adversarial training (AT) (Madry et al. 2018; Zhang et al. 2019; Ma, Wang, and Liu 2022; Li and Liu 2023; Wang et al. 2023), one of the most effective and widely-used approaches among adversarial defenses, improves the robustness of NN by augmenting the training set with adversarial examples (Figure 1a). AT-based RS can be viewed as a robust optimization process that involves seeking a pointwise adversary of the smoothed classifier, which generates the worst-case adversarial example by independently perturbing each benign image.

Note that previous works of AT-based RS focus on *pointwise* certification, *i.e.*, trying to provide a provable guarantee for the worst-case adversarial example around the given input, as shown in Figure 1b. However, these works ignore the performance of the smoothed classifier on the input data population, which is an important indicator, since it relates

*Corresponding author.



(a) Adversarial Training (AT) (b) AT-based RS (c) DRF (Ours)

Figure 1: Illustrations of decision boundaries obtained by different training methods. (a) Adversarial training (Madry et al. 2018) corrects adversarial examples found within an ϵ-ball around each sample. (b) AT-based RS methods (Salman et al. 2019; Jeong and Shin 2020) employ AT on smoothed classifiers to provide certified robustness, resulting in smoother decision boundaries compared to AT. However, these methods consistently output decisions within the neighbourhood of input with a **fixed** attack strength ϵ, leading to reduced generalization capacity. (c) DRF (ours) generates adversarial examples by considering the data distribution, which allows alteration of the maximum distance between the adversarial example and input. It is novel to remove the hard constraint since randomized smoothing is a general defense regardless of specific ϵ.

directly to the *generalization capacity* of the smooth classifier. Accordingly, the following question is raised:

- Can we design a training strategy that considers the data distribution during adversarial training, in order to improve the performance of the smoothed classifier?

To fill this gap, our work attempts to apply distributional robustness (DR) to improve the generalization capacity. Specifically, we propose a novel framework to connect AT-based RS methods with DR, namely, a Distributional Robustness Framework for smoothed classifiers (DRF). DR (Sinha, Namkoong, and Duchi 2018; Blanchet and Murthy 2019; Bui et al. 2022; Zhou and Liu 2023) looks for a worst-case distribution located in the ball centered around the data distribution, which generates the adversarial examples. Therefore, compared with AT, the adversary in DR does not look for the perturbation of a specific data example, but rather moves the worst-case distribution around the data distribution. Thus, DRF is expected to achieve better generalization than the original AT-based RS methods on unseen data (Figure 1c). Our main contributions can be summarized as follows:

1. Theoretically, our proposed DRF bridges AT-based randomized smoothing and distributional robustness. By adopting this approach, we can generalize and encompass AT-based RS methods in the DR setting, including SmoothAdv (Salman et al. 2019) and Consistency (Jeong and Shin 2020). We prove that these methods are special cases of their DR counterparts. Motivated by theoretical analysis, we further develop a novel algorithm to unify the AT-based RS, which has better generalization capacity than the original method.
2. Empirically, we evaluate DRF against various robust training methods for RS. The results consistently show that our framework significantly improves the certified robustness compared to existing methods. Through sensitivity analysis, we further verify that our method offers a new, stable trade-off term between the clean accuracy and certified robustness, and also performs stably in training for a wide range of hyperparameters.

2 Preliminaries

Distributional robustness (DR). We consider the classical stochastic optimization problem, given a generic Polish space $\mathcal{Z}$ endowed with a data distribution $\mathcal{P}$; $\ell_\theta : \mathcal{Z} \to \mathbb{R}$ is a loss function over a parameter $\theta \in \Theta$, $c : \mathcal{Z} \times \mathcal{Z} \to \mathbb{R}_+$ is a cost function. The distributional robustness setting (Sinha, Namkoong, and Duchi 2018; Blanchet and Murthy 2019; Bui et al. 2022) aims to find the distribution $\mathcal{Q}$ in the vicinity of $\mathcal{P}$ and maximize the loss in the expectation, then certify the performance even for the worst-case population loss, which requires solving the following min-max problem to attain a robust classifier:

$$\inf_{\theta} \sup_{\mathcal{Q}:\mathcal{W}_c(\mathcal{P},\mathcal{Q})<\epsilon} \mathbb{E}_{\mathcal{Q}} \left[\ell_\theta (z)\right], \tag{1}$$

where $\epsilon > 0$ and $z \sim \mathcal{Q}$. $\mathcal{W}_c$ denotes the optimal transport (OT) cost, or a Wasserstein distance if c is a metric. Wasserstein distance defines a notion of closeness between distributions as follows:

$$\mathcal{W}_c (\mathcal{P}, \mathcal{Q}) := \inf_{\pi \in \Pi(\mathcal{P},\mathcal{Q})} \int c \, d\pi, \tag{2}$$

where $\Pi (\mathcal{P}, \mathcal{Q})$ is the set of couplings whose marginals are $\mathcal{P}$ and $\mathcal{Q}$, the cost c is a non-negative lower semi-continuous function that satisfies $c(z, z) = 0$. Sinha, Namkoong, and Duchi (2018); Blanchet and Murthy (2019); Bui et al. (2022) show that the *strong duality* holds:

$$\begin{aligned} &\sup_{\mathcal{Q}:\mathcal{W}_c(\mathcal{P},\mathcal{Q})<\epsilon} \mathbb{E}_{\mathcal{Q}} \left[\ell_\theta (z)\right] \\ &= \inf_{\lambda \geq 0} \left\{ \lambda\epsilon + \mathop{\mathbb{E}}_{z\sim\mathcal{P}} \left[\sup_{z' \in \mathcal{Z}} \left\{ \ell_\theta (z') - \lambda c (z', z) \right\} \right] \right\}. \end{aligned} \tag{3}$$

Local adversarial robustness. We consider the classification task with K classes. Let $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$, $z = (x, y) \in \mathcal{Z}$,

where $x \in \mathcal{X}$ and $y \in \mathcal{Y} \coloneqq \{1, \cdots, K\}$ denote an input and the corresponding class label, respectively. Let $f_\theta : \mathcal{X} \to \mathcal{Y}$ be the classifier with parameter $\theta \in \Theta$, which is modeled by $f_\theta(x) \coloneqq \arg\max_{k \in \mathcal{Y}} F_\theta^k(x)$ with a differentiable mapping $F_\theta : \mathcal{X} \to \Delta^{K-1}$, where Δ^{K-1} denotes the probability simplex in $\mathbb{R}^K$. In this paper, F_θ is an NN with parameter $\theta \in \Theta$ followed by a softmax layer.

In the context of *local adversarial robustness* for NNs, we require f_θ not only to correctly classify $(x, y) \sim \mathcal{P}$, but also to be locally constant around x; *i.e.*, f_θ is certified not to contain any adversarial examples in the L_2 ball centered at x. Accordingly, one can measure the adversarial robustness of a classifier f_θ by considering the largest possible radius of the L_2 ball (also called the *robust radius*), defined as follows:

$$R(f_\theta; x, y) \coloneqq \min_{f_\theta(x') \neq y} \|x' - x\|_2. \tag{4}$$

Therefore, our goal is to train an f_θ that performs well on $\mathcal{P}$, while also maximizing $R(f_\theta; x, y)$.

Randomized smoothing (RS). Unfortunately, computing the *robust radius* (Eq. (4)) is proven to be an NP-complete problem (Katz et al. 2017; Sinha, Namkoong, and Duchi 2018). In cases when f_θ is too complex to control its predictions in practice (*e.g.*, if f_θ is a large NN on high-dimensional data), solving Eq. (4) directly will be time-consuming. *Randomized smoothing* (Cohen, Rosenfeld, and Kolter 2019) instead constructs a new classifier $\hat{f}_\theta$ from f_θ that can more easily obtain robustness by "smoothly" transforming the base classifier f_θ with the Gaussian distributions $\mathcal{N}(0, \sigma^2 I)$:

$$\hat{f}_\theta(x) \coloneqq \arg\max_{c \in \mathcal{Y}} \mathbb{P}_{\delta \sim \mathcal{N}(0, \sigma^2 I)} \left(f_\theta(x + \delta) = c \right), \tag{5}$$

where σ^2 is a hyperparameter that controls the level of smoothing. For a given (x, y), Cohen, Rosenfeld, and Kolter (2019) show that $R(\hat{f}_\theta; x, y)$ can be lower-bounded by the *certified radius* $\underline{R}(\hat{f}_\theta, x, y)$; this can be derived from the *confidence* of $\hat{f}_\theta$ at x, which we denote by $p_f(x)$, as follows:

$$\underline{R}(\hat{f}_\theta, x, y) \coloneqq \sigma \cdot \Phi^{-1}(p_f(x)) \leq R(\hat{f}_\theta; x, y), \tag{6}$$

where $p_f(x) \coloneqq \mathbb{P}_{\delta \sim \mathcal{N}(0, \sigma^2 I)}(f_\theta(x + \delta) = y)$, provided that $\hat{f}_\theta(x) = y$; otherwise, $R(\hat{f}_\theta; x, y) \coloneqq 0$. Φ denotes the cumulative distribution function of the standard normal distribution. This lower bound is known to be tight for the L_2-minimum distance, *e.g.*, the bound $\underline{R}$ is optimal for linear classifiers (Cohen, Rosenfeld, and Kolter 2019).

Adversarial examples for smoothed classifiers. Salman et al. (2019) were the first to present a direct attack for smoothed classifiers. It is difficult to directly find adversarial examples for $\hat{f}_\theta$ because of the argmax; besides, $\hat{f}_\theta$ is essentially a non-differentiable object when Eq. (5) is approximated via Monte Carlo sampling (Metropolis and Ulam 1949). Thus, SmoothAdv proposes to attack the *soft-smoothed* classifier $\hat{F}_\theta \coloneqq \mathbb{E}_\delta[F_\theta(x + \delta)]$ rather than $\hat{f}_\theta$, as $\hat{F}_\theta : \mathbb{R}^d \to \Delta^{K-1}$ is differentiable. Specifically, SmoothAdv finds an adversarial example $\hat{x}$ under attack strength ϵ:

$$\begin{aligned} \hat{x} &= \arg\max_{\|x' - x\|_2 \leq \epsilon} \mathrm{CE}(\hat{F}_\theta(x' + \delta), y) \\ &= \arg\max_{\|x' - x\|_2 \leq \epsilon} \left(-\log \mathbb{E}_\delta \left[F_\theta^y(x' + \delta) \right] \right), \end{aligned} \tag{7}$$

where CE denotes the standard cross-entropy loss. In practice, the expectation in this objective Eq. (7) is approximated via Monte Carlo integration with m samples of δ, namely $\delta_1, \cdots, \delta_m \sim \mathcal{N}(0, \sigma^2 I)$:

$$\hat{x} = \arg\max_{\|x' - x\|_2 \leq \epsilon} \left(-\log \left(\frac{1}{m} \sum_i F_\theta^y(x' + \delta_i) \right) \right). \tag{8}$$

Adversarial training (AT) for smoothed classifiers. To improve the robustness of $\hat{f}_\theta$ when smoothing with Gaussian noise, (Cohen, Rosenfeld, and Kolter 2019) simply proposes to train f_θ using Gaussian augmentation:

$$\inf_\theta \mathop{\mathbb{E}}_{\substack{(x,y) \sim \mathcal{P} \\ \delta \sim \mathcal{N}(0, \sigma^2 I)}} \left[\mathrm{CE}(F_\theta(x + \delta), y) \right]. \tag{9}$$

To obtain an f_θ that gives a more robust classifier when smoothed into $\hat{f}_\theta$, previous works (Salman et al. 2019; Jeong and Shin 2020) adopt adversarial training on $\hat{f}_\theta$ using the adversarial examples generated by Eq. (8).

SmoothAdv (Salman et al. 2019) employs adversarial training with PGD attack (Madry et al. 2018) on the smoothed classifiers:

$$\inf_\theta \mathbb{E}_{\mathcal{P}} \left[\sup_{\|x' - x\|_2 \leq \epsilon} \mathbb{E}_\delta \left[\mathrm{CE}(F_\theta(x' + \delta), y) \right] \right]. \tag{10}$$

Consistency (Jeong and Shin 2020) applies a *consistency regularization* term to Eq. (10):

$$\begin{aligned} \inf_\theta \mathbb{E}_{\mathcal{P}} \Bigg[\sup_{\|x' - x\|_2 \leq \epsilon} \mathbb{E}_\delta \big[&\mathrm{CE}(F_\theta(x' + \delta), y) \\ &+ \lambda \cdot \mathrm{KL}(\hat{F}_\theta(x') || F_\theta(x' + \delta)) + \eta \cdot \mathrm{H}(\hat{F}_\theta(x')) \big] \Bigg], \end{aligned} \tag{11}$$

where $\mathrm{KL}(\cdot||\cdot)$ and $\mathrm{H}(\cdot)$ denote the Kullback–Leibler (KL) divergence and the entropy, respectively, while $\lambda, \eta > 0$ are hyperparameters that control the relative strength.

3 DRF: Distributional Robustness Framework for Smoothed Classifiers

In this section, we propose a unified formulation for distributional robustness, which is a more general framework for connecting AT-based RS methods and existing distributional robustness approaches.

3.1 Theoretical Framework

In this subsection, we propose a unified formulation for distributional robustness, which is a more general framework for connecting AT-based RS methods and existing distributional robustness approaches.

Given the data distribution $\mathcal{P}^{data}$, let the input $x \sim \mathcal{P}^{data}$. $\mathcal{P}^{label}_{\cdot|x}$ is the conditional distribution for a given x, used to generate label $y \sim \mathcal{P}^{label}_{\cdot|x}$. We consider the space $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$; the joint distribution $\mathcal{P}_\triangle$ on $\mathcal{Z}$ consists of samples (x, y), where $x \sim \mathcal{P}^{data}$ and $y \sim \mathcal{P}^{label}_{\cdot|x}$.

Consider a distribution $\mathcal{Q}$ on $\mathcal{Z}$ that satisfies the condition $\mathcal{W}_c(\mathcal{P}_\triangle, \mathcal{Q}) < \epsilon$. A sample z drawn from $\mathcal{P}_\triangle$ is represented as $z = (x, y)$, while a sample z' from $\mathcal{Q}$ is denoted as $z' = (x', y')$. We define a unified loss function $\mathcal{L}_\theta(z', \delta) : \mathcal{Z} \times \mathcal{X} \to \mathbb{R}$ disturbed by δ over a parameter $\theta \in \Theta$. We can then transform *SmoothAdv* (Eq. (10)) and *Consistency* (Eq. (11)) into their distributional robustness versions, as follows:

- *SmoothAdv* (Salman et al. 2019): $\mathcal{L}_\theta(z', \delta) := \mathrm{CE}(F_\theta(x' + \delta), y')$.
- *Consistency* (Jeong and Shin 2020): $\mathcal{L}_\theta(z', \delta) := \mathrm{CE}(F_\theta(x' + \delta), y') + \lambda \cdot \mathrm{KL}(\hat{F}_\theta(x') || F_\theta(x' + \delta)) + \eta \cdot \mathrm{H}(\hat{F}_\theta(x'))$.

Remark 1. In this setting, x' tends to be close to x. Since $\mathcal{P}^{data}$ is the marginal distribution of $\mathcal{P}_\triangle$ on x, if $\mathcal{Q}^{data}$ is the marginal distribution of $\mathcal{Q}$ on x' in (x', y'), then $\mathcal{W}_c\left(\mathcal{P}^{data}, \mathcal{Q}^{data}\right) \le \mathcal{W}_c(\mathcal{P}_\triangle, \mathcal{Q}) < \epsilon$, which explains the closeness between x and x'. This property is necessary to the local adversarial robustness, since the adversary can usually engineer x' so that it appears identical to x to the human eye.

Remark 2. $\mathbb{E}_\delta[\mathcal{L}_\theta(z', \delta)]$ can be seen as a smoothed version of $\mathcal{L}_\theta(z', \delta)$; moreover, since the normal distribution has a density, the smoothed function is guaranteed to be differentiable (Bertsekas 1973).

Then we prove the equivalence of the primal and dual forms for the smoothed loss function in the following theorem. The formal version and corresponding proof are presented in full in Appendix.

Theorem 1 (informal)**.** *Assume that the loss function $\mathcal{L}_\theta$ is upper semi-continuous and the cost function c is lower semi-continuous. We then have the following equality between the primal form and dual form:*

$$\sup_{\mathcal{Q}:\mathcal{W}_c(\mathcal{P}_\triangle,\mathcal{Q})<\epsilon} \mathbb{E}_{\mathcal{Q},\delta}\left[\mathcal{L}_\theta(z', \delta)\right] = \inf_{\lambda \ge 0}\left\{\lambda\epsilon + \mathop{\mathbb{E}}_{z\sim\mathcal{P}_\triangle}\left[\sup_{z'\in\mathcal{Z}}\left\{\mathbb{E}_\delta[\mathcal{L}_\theta(z', \delta)] - \lambda c(z', z)\right\}\right]\right\}. \tag{12}$$

Remark 3. There are two unique challenges we need to overcome in the proof: (a) The first challenge is the incorporation of the noise level δ in smoothed loss $\mathbb{E}_\delta[\mathcal{L}_\theta(z', \delta)]$, since we need to consider the expectation of this additional random variable. Notably, the smoothed loss in DR was hitherto unexplored. (b) In DRF, we impose a substantial penalty on points outside the epsilon ball, requiring the infinity term in the cost function c. This contrasts with previous distributional robustness work: the primal-dual form in Eq. (3) employs a bounded cost function. In Theorem 1, we bypass the restrictive assumption and prove the primal-dual transformation for the unlimited cost function, tailored for randomized smoothing scenarios.

Algorithm 1: DRF training

Input: Sample $(x, y) \sim P$, smoothing factor σ, number of noise samples m, current iteration n, number of steps T, step size of adversarial example α, initial value of regularization λ_0, learning rate ζ_λ of λ, learning rate ζ_θ of θ, batch size of adversarial examples κ, attack strength ϵ.

Sample $\delta_1, \cdots, \delta_m \sim \mathcal{N}(0, \sigma^2 I)$
// FIND AN ADVERSARIAL EXAMPLE
$\hat{x}^{(0)}, \hat{F}_\theta(x^{(0)}) \leftarrow x, \frac{1}{m}\sum_{i=1}^m F_\theta(x + \delta_i)$
for $t = 0$ **to** $T - 1$ **do**
  $\hat{x}^{(t+1)}_{\text{inter}} \leftarrow \hat{x}^{(t)} + \alpha \nabla_x(-\log \hat{F}^y_\theta(\hat{x}^{(t)}))$
  $\hat{x}^{(t+1)} \leftarrow \hat{x}^{(t+1)}_{\text{inter}} - \lambda_n \|\hat{x}^{(t+1)}_{\text{inter}} - x\|_2$
  $\hat{x}^{(t+1)} \leftarrow \texttt{clip}(\hat{x}^{(t+1)}, 0, 1)$
  // CLIP TO VALID RANGE
  $\hat{F}(\hat{x}^{(t+1)}) \leftarrow \frac{1}{m}\sum_{i=1}^m F(\hat{x}^{(t+1)} + \delta_i)$
end for
// UPDATE MODEL PARAMETER θ
$\theta_{n+1} \leftarrow \theta_n - \frac{\zeta_\theta}{m}\sum_i \nabla_\theta \mathcal{L}_\theta(\hat{x}^{(T)}, y, \delta_i)$
// UPDATE REGULARIZATION STRENGTH λ
$\lambda_{n+1} = \lambda_n - \zeta_\lambda\left(\epsilon - \frac{1}{\kappa}\sum_{i=1}^\kappa \|\hat{x}_i - x_i\|_2\right)$

3.2 Implementation for DRF

In this subsection, we introduce the specific implementation of our proposed DRF. See pseudocode in Algorithm 1. RS focuses on the certified robustness in L_2 space; thus, we consider the Euclidean norm $\|\cdot\|_2$ as the cost function, which is differentiable except at the null point. Then, the final optimized objective of DRF becomes:

$$\inf_{\theta,\lambda\ge 0}\left\{\lambda\epsilon + \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{P}_\triangle}\left[\sup_{x'\in\mathcal{X}}\left\{\mathbb{E}_\delta\left[\mathcal{L}_\theta(x', y, \delta)\right] - \lambda\|x' - x\|_2\right\}\right]\right\}. \tag{13}$$

Similar to the original AT-based RS methods, we employ iterative gradient ascent to find adversarial example $\hat{x}$. Given the current parameter λ_n (*i.e.*, the initial value λ_0 after updating for n iterations), we optimize the following objective modified by Eq. (7) to obtain $\hat{x}$:

$$\hat{x} = \arg\max_{x'}\left\{\mathrm{CE}(\hat{F}_\theta(x' + \delta), y) - \lambda_n\|x' - x\|_2\right\}. \tag{14}$$

Comparing Eq. (14) with original RS attacker (Eq. (7)), we do not apply any explicit project operation into the ball $\mathcal{B}_\epsilon(x)$ (*e.g.*, RENORM in PyTorch). The project procedure of DRF can be seen as "soft projection", *i.e.*, implicitly projecting into a soft ball governed by λ. The adversarial example $\hat{x}$ gets closer to x when the value of λ increases.

Then, we update the model parameter θ in order to optimize the outer minimization objective in Eq. (13). Following previous works (Salman et al. 2019; Jeong and Shin 2020), we minimize the averaged loss over $(\hat{x} + \delta_1, y), \cdots, (\hat{x} + \delta_m, y)$, *i.e.*, $\min_\theta \frac{1}{m}\sum_i \mathcal{L}_\theta(\hat{x}, y, \delta_i)$, where $\delta_1, \cdots, \delta_m \sim \mathcal{N}(0, \sigma^2 I)$; different methods specify $\mathcal{L}_\theta$ differently (see Section 3.1).

Finally, we update the coefficient of the regularization term λ. Given the batch size κ , we preserve a batch of ad-

versarial examples $\{\hat{x}_i\}_{i=1}^{\kappa}$ and corresponding benign examples $\{x_i\}_{i=1}^{\kappa}$. We define Δ_κ as the average distance from $\hat{x}$ to x, *i.e.*, $\Delta_\kappa = \frac{1}{\kappa}\sum_{i=1}^{\kappa} \|\hat{x}_i - x_i\|_2$. If Δ_κ is less than ϵ, meaning that adversarial examples are globally close to benign examples, then λ should be decreased; otherwise, λ should be increased because of $\hat{x}$ being far from x globally. Thus, given the current λ_n and learning rate ζ_λ, we update $\lambda_{n+1} = \lambda_n - \zeta_\lambda(\epsilon - \Delta_\kappa)$.

Remark 4. The intuition behind learnable λ is straightforward: using a fixed λ to perturb diverse images can be suboptimal. When λ is small, the resulting adversarial examples are far from their benign counterparts, making classification difficult. Conversely, setting λ to a high value makes the adversarial examples similar to benign examples, rendering the model vulnerable to adversarial attacks. Thus, we employ a learnable λ to achieve better generalization performance. Our approach leverages the concept of softball to generate diverse and flexible adversarial examples, some of which lie within the ϵ-balls while others lie outside. It is natural to remove the "hard projection" since randomized smoothing is a general defense regardless of specific attack strength ϵ.

3.3 Efficacy of DRF

To demonstrate the efficacy of DRF, we depict the confidence gap between the ground truth and the "runner-up" class across Gaussian noise samples $(x+\delta, y)$, defined as $\log F_\theta^y(x+\delta) - \max_{c\neq y} \log F_\theta^c(x+\delta)$. Intuitively, samples that are easily and accurately classified exhibit a large confidence gap, thereby yielding a large certified radius (Eq. (6)).

In Figure 2, for each frequency distribution histogram and kernel density estimation (KDE), we utilize noise level $\sigma = 1.0$ and the complete test set (10,000 samples) of CIFAR-10, aligning with the results of $\sigma = 1.0$ in Appendix. We exclusively count samples that are classified correctly. The observations are as follows:

- The confidence gap in the *Consistency* is notably lower and more concentrated compared to the *SmoothAdv*. This indicates that both easy and challenging samples exhibit a consistently robust radius. As a result, *Consistency* underperforms at lower values of r but surpasses *SmoothAdv* at larger r.
- This phenomenon arises due to the regularization term $\mathrm{KL}(\hat{F}_\theta(x')\|F_\theta(x'+\delta))$ within the loss function of *Consistency* (Eq. (11)). This term enforces alignment between the confidence of the clean sample ($r = 0$) and the confidence of the noisy sample. However, such over-regularization can lead to poor performance for small values of r. In contrast, *SmoothAdv* does not distinctly focus on the "hard" samples ($r = \sigma$), which results in a prolonged tail in its histogram.

Figure 2 further unveils the following insights into our method. By regulating the distance between the adversarial example and its benign counterpart using the learnable λ:

- Our DRF mitigates the heavy-tailed issue within the *SmoothAdv* histogram.
- For *Consistency*, DRF effectively moderates over-regularization, leading to a less concentrated histogram.

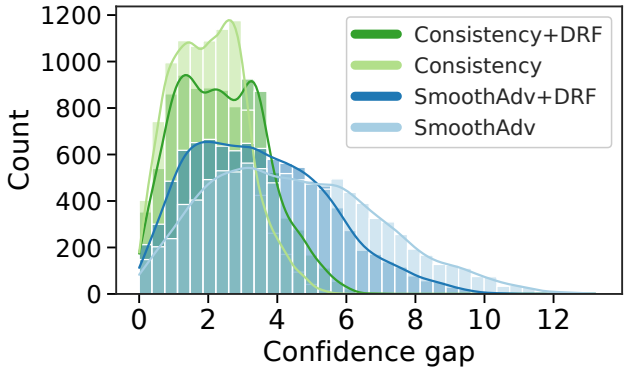


Figure 2: Frequency distribution histogram and kernel density estimation (KDE) for the confidence gap, quantified as $\log F_\theta^y(x+\delta) - \max_{c\neq y} \log F_\theta^c(x+\delta)$.

These experimental results verify the efficacy of our approach through the incorporation of a learnable λ. Through the dynamic adjustment of the distance between adversarial examples and their benign counterparts via adaptive λ, we can achieve a more harmonious balance between clean accuracy and certified robustness and result in an enhanced Average Certified Radius (ACR).

4 Experiments

In this section, we evaluate the effectiveness of our framework on well-established image classification datasets to measure robustness, including MNIST (LeCun et al. 1998) and CIFAR-10 (Appendix) (Krizhevsky and Hinton 2009). Additionally, we conduct sensitivity analysis in Section 4.3 to further investigate the components in our framework. Details of the experimental setup (*e.g.*, datasets, computing resources, hyperparameters for the baseline methods, etc.) are provided in Appendix.

4.1 Setups

Baseline methods. We compare our framework with a variety of existing techniques proposed for the robust training of smoothed classifiers: ❶ *Gaussian-based*: (a) Gaussian (Cohen, Rosenfeld, and Kolter 2019): standard training with Gaussian augmentation; (b) Stability training (Li et al. 2019): a cross-entropy regularization between $F_\theta(x)$ and $F_\theta(x+\delta)$; (c) MACER (Zhai et al. 2020): a regularization that maximizes an approximative form of the certified radius in Eq. (6); (d) SmoothMix (Jeong et al. 2021): uses mixup loss to regularize the over-confident predictions; ❷ *AT-based*: (e) SmoothAdv (Salman et al. 2019): adversarial training on the smoothed classifier; (f) Consistency (Jeong and Shin 2020): a KL-divergence-based regularization that minimizes the variance of $F_\theta(x+\delta)$ across different δ.

Training details. We train every model via stochastic gradient descent (SGD) using Nesterov momentum of weight 0.9 without dampening. The weight decay and batch size are set to 10^{-4} and 256 for all the models. The different training schedules for each dataset are provided below:

σ	Models (MNIST)	ACR	0.00	0.25	0.50	0.75	1.00	1.25	1.50	1.75	2.00	2.25	2.50	2.75
0.25	Gaussian (Cohen, Rosenfeld, and Kolter 2019)	0.911	99.2	98.4	96.7	93.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	Stability training (Li et al. 2019)	0.915	99.3	98.6	97.1	93.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	MACER (Zhai et al. 2020)	0.920	99.3	98.7	97.4	94.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	SmoothMix (Jeong et al. 2021)	0.927	99.4	98.9	97.9	96.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	SmoothAdv (Salman et al. 2019)	0.932	99.4	98.9	98.1	96.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	+ DRF (ours)	**0.933**	**99.4**	**99.0**	**98.2**	**96.9**	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	Consistency (Jeong and Shin 2020)	0.932	99.3	98.9	98.1	96.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	+ DRF (ours)	**0.933**	**99.3**	**99.0**	**98.2**	**97.0**	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.50	Gaussian (Cohen, Rosenfeld, and Kolter 2019)	1.555	99.1	98.3	96.9	94.3	89.7	81.8	67.5	44.3	0.0	0.0	0.0	0.0
	Stability training (Li et al. 2019)	1.571	99.2	98.5	97.2	94.9	90.7	83.0	69.2	45.9	0.0	0.0	0.0	0.0
	MACER (Zhai et al. 2020)	1.598	98.7	98.0	96.4	94.0	90.2	83.7	72.4	54.0	0.0	0.0	0.0	0.0
	SmoothMix (Jeong et al. 2021)	1.677	99.1	98.5	97.5	95.7	92.9	88.2	80.0	65.2	0.0	0.0	0.0	0.0
	SmoothAdv (Salman et al. 2019)	1.688	99.0	98.3	97.3	95.7	93.0	88.6	81.1	67.6	0.0	0.0	0.0	0.0
	+ DRF (ours)	**1.693**	99.0	**98.4**	**97.4**	**95.8**	**93.2**	**88.7**	**81.6**	**68.2**	0.0	0.0	0.0	0.0
	Consistency (Jeong and Shin 2020)	1.692	98.6	98.0	97.0	95.3	92.5	88.3	81.8	70.3	0.0	0.0	0.0	0.0
	+ DRF (ours)	**1.697**	98.5	98.0	**97.0**	**95.2**	**92.6**	**88.4**	**82.0**	**70.8**	0.0	0.0	0.0	0.0
1.00	Gaussian (Cohen, Rosenfeld, and Kolter 2019)	1.620	96.3	94.4	91.4	86.8	79.8	70.9	59.4	46.2	32.5	19.7	10.9	5.8
	Stability training (Li et al. 2019)	1.631	96.4	94.6	91.6	87.2	80.6	71.8	60.5	46.9	33.1	20.0	11.2	5.7
	MACER (Zhai et al. 2020)	1.593	91.6	88.1	83.5	77.7	71.3	63.7	55.6	47.3	38.4	29.2	20.0	11.5
	SmoothMix (Jeong et al. 2021)	1.786	95.6	93.6	90.5	86.3	80.6	73.5	64.3	53.7	43.1	33.2	23.7	13.7
	SmoothAdv (Salman et al. 2019)	1.776	95.7	93.9	90.6	86.5	80.8	73.6	64.4	53.8	43.2	32.8	22.2	12.0
	+ DRF (ours)	**1.788**	95.6	93.7	90.6	86.3	**80.7**	**73.6**	**64.5**	**54.1**	**44.1**	**34.1**	**24.2**	**13.8**
	Consistency (Jeong and Shin 2020)	1.818	94.2	92.0	88.5	84.3	79.0	72.1	63.8	54.6	45.4	37.1	28.0	19.4
	+ DRF (ours)	**1.824**	94.1	92.0	88.5	84.2	78.9	**72.2**	**63.9**	**54.8**	**45.7**	**37.5**	**28.5**	**20.0**

Table 1: Comparison of the certified accuracy (%) and ACR on MNIST. Every model is certified with σ used for its training. Each value except ACR indicates the fraction of test samples with an L_2 certified radius larger than the threshold specified in the top row. We highlight our results in bold whenever the value improves relative to the Gaussian baseline, and underline them whenever the value represents an improvement relative to the considered AT-based RS baseline.

(a) MNIST: The initial learning rate is set to 0.01. We train LeNet (LeCun et al. 1998) model for 90 epochs. (b) CIFAR-10: The initial learning rate is set to 0.1; We train ResNet-110 (He et al. 2016) model for 150 epochs, and the learning rate is decayed by 0.1 at 50-th and 100-th epoch.

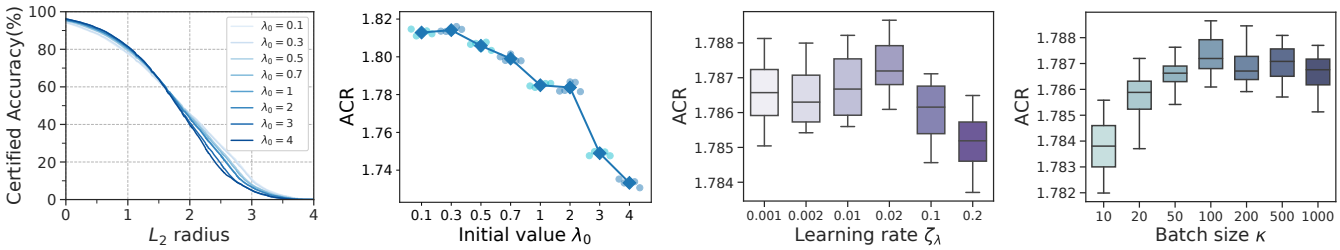
Evaluation metrics. Our evaluation of the robustness is based on the practical certification procedure CERTIFY proposed by Cohen, Rosenfeld, and Kolter (2019) and similar to a number of prior works (Salman et al. 2019; Zhai et al. 2020; Jeong and Shin 2020; Jeong et al. 2021). We consider two evaluation metrics: (a) the *certified accuracy* at various radii, which is the fraction of the test dataset that CERTIFY classifies correctly, and (b) the *average certified radius* (ACR) (Zhai et al. 2020), namely the average of certified radii returned by CERTIFY on the test dataset counting only the correctly classified samples, *i.e.*, $\mathrm{ACR} := \frac{1}{|\mathcal{D}_{\mathtt{test}}|} \sum_{(x,y)\in\mathcal{D}_{\mathtt{test}}} \mathrm{CR}(f_\theta, \sigma, x) \cdot \mathbb{1}_{[\hat{f}(x)=y]}$, where $\mathcal{D}_{\mathtt{test}}$ is the test dataset and CR is the certified radius returned by CERTIFY (Eq. (6)). We apply CERTIFY with $n = 100,000$, $n_0 = 100$ and $\alpha = 0.001$, following (Cohen, Rosenfeld, and Kolter 2019; Salman et al. 2019; Jeong and Shin 2020; Jeong et al. 2021).

Remark 5. The certified accuracy is a function of the fixed attack strength r; it is difficult to compare the robustness of two models unless one is uniformly better than the other for all values of r. Thus, ACR is a more suitable choice than certified accuracy for comparing the robustness under conditions of trade-off between accuracy and robustness (Tsipras et al. 2019; Zhang et al. 2019). By its definition, ACR naturally assigns 0 for the incorrectly classified test samples, *i.e.*, $\hat{f}_\theta(x) \neq y$; thus, a decreasing clean accuracy of $\hat{f}_\theta$ would negatively affect the value of ACR.

4.2 Results on MNIST

For the MNIST (LeCun et al. 1998) experiments, we train every method on LeNet (LeCun et al. 1998) for 90 epochs, then report the certified accuracy and ACR of smoothed classifiers using the full MNIST test dataset. When DRF is used, we employ fixed hyperparameter values of $\zeta_\lambda = 0.02$ and $\kappa = 100$ in this subsection, while $\lambda_0 = 1.0, 2.0$ for SmoothAdv and Consistency, respectively. For the other hyperparameters of DRF added to these two methods, we utilize the same configurations as outlined in the original works (Salman et al. 2019; Jeong and Shin 2020). See Appendix for the specific hyperparameters of these two baselines.

For each model configuration, we train and evaluate the models with noise levels $\sigma \in \{0.25, 0.5, 1.0\}$, the results are reported in Table 1. Due to space limitations, figures of the certified accuracy over the full range of radii per σ are attached in Appendix. The improvements of our proposed DRF are summarized below:



(a) Effect of λ_0 on accuracy (b) Effect of λ_0 on ACR (c) Effect of ζ_λ (d) Effect of κ

Figure 3: Comparison of the certified accuracy and ACR across models with various parameters via DRF. ACR results are reported over five runs on MNIST.

- DRF consistently improves "SmoothAdv" and "Consistency" baselines in terms of ACR. DRF obtains a new SOTA ACR compared with all baselines. These results verifiy an orthogonal contribution of our DRF compared to prior works.
- DRF generally exhibits a better trade-off between clean accuracy and robustness: *e.g.*, for $\sigma = 1.0$, the clean accuracy declines only slightly, while ACR and the certified accuracy at large r are significantly promoted. When $\sigma = 0.5$, DRF can effectively preserve the clean accuracy while also improving the certified accuracy at large r. Overall, our DRF can further improve the certified accuracy, especially at large radii, and achieve better performance in ACR.
- Compared with the notable increase introduced by DRF relative to the "Gaussian" baseline, the improvements of other regularization-based approaches (*e.g.*, Stability training, MACER) are limited. This is because the regularization term λ in DRF is adaptable; it can leverage the global information when solving the outer minimization in Eq. (13).

4.3 Sensitivity Analysis

In this section, we conduct a sensitivity analysis on MNIST to carefully examine the effect of different hyperparameters in DRF. We perform experiments on "SmoothAdv + DRF" with $\sigma = 1.0$.

The initial value of λ. By design, λ controls the distance between the adversarial example $\hat{x}$ and original input x. λ is adaptive and updated by Algorithm 1. We further examine the effect of λ_0 (*i.e.*, the initial value of λ) on two metrics:

- **Certified accuracy:** In Figure 3a, we illustrate the certified accuracy with respect to (w.r.t) varying λ_0 when $\sigma = 1.0$. As expected, we observe a clear trade-off between the clean accuracy and the certified robustness. When λ_0 increases, the clean accuracy of the smooth classifier increases while the certified accuracy at large r decreases. This reveals that λ significantly impacts the performance of the smoothed classifiers as an effective term to trade off the robustness against the clean accuracy. Thus, it is necessary for us to find a proper trade-off between the clean accuracy and robustness; to do so, we choose $\lambda_0 = 1.0$ for MNIST in Section 4.2.
- **ACR:** Figure 3b plots how ACR changes w.r.t λ_0. From the figure, we can see that ACR decreases as λ_0 increases, which reveals that we can control ACR by adjusting λ_0. Recall that as λ_0 increases, the clean accuracy of the smooth classifier increases. This is because an increase in the clean accuracy often makes less of a contribution to the increase in ACR, *i.e.*, $\hat{f}$ correctly classifies more test samples with the CR closer to 0.

Effect of ζ_λ. In DRF, the learning rate ζ_λ of λ is a hyperparameter. To determine whether the performance of DRF is sensitive to ζ_λ, we conduct experiments on the ACR w.r.t different values of ζ_λ and present the results in Figure 3c. As shown in Figure 3c, the performance of DRF is not very sensitive to ζ_λ when it is within the range of 0.001 to 0.02, then suffers from a small drop when ζ_λ is larger than 0.02. Thus we choose $\zeta_\lambda = 0.02$ for MNIST in Section 4.2. The results demonstrate that DRF can work well over a wide range of learning rate ζ_λ.

Effect of κ. We investigate the influence on ACR of the batch size of adversarial examples κ to update λ (see Figure 3d). From the result, we can determine that ACR increases slightly with rising κ, then remains stable at values of κ larger than 100; thus, we choose $\kappa = 100$ for the main experiments. The results illustrate that the performance of the smoothed classifier is not sensitive to κ.

5 Conclusion

In this paper, we view certified robustness from a different perspective, *i.e.*, the worst-case population loss over the input distribution. We provide a novel unified distributional robustness framework for randomized smoothing, namely DRF. Our proposed approach unifies and improves the performance of the selected AT-based RS methods, namely SmoothAdv and Consistency, and achieves SOTA results compared with various baselines.

Acknowledgements

This work is supported by the National Key R&D Program of China under Grant 2023YFC3604702, the National Natural Science Foundation of China under Grant 61976161, the Fundamental Research Funds for the Central Universities under Grant 2042022rc0016.

References

Bertsekas, D. P. 1973. Stochastic optimization problems with nondifferentiable cost functionals. *Journal of Optimization Theory and Applications*, 12(2): 218–231.

Blanchet, J. H.; and Murthy, K. R. A. 2019. Quantifying Distributional Model Risk via Optimal Transport. *Mathematics of Operations Research*, 44(2): 565–600.

Bui, T. A.; Le, T.; Tran, Q. H.; Zhao, H.; and Phung, D. Q. 2022. A Unified Wasserstein Distributional Robustness Framework for Adversarial Training. In *ICLR*.

Carlini, N.; and Wagner, D. A. 2017. Towards Evaluating the Robustness of Neural Networks. In *SP*, 39–57.

Chen, Y.; and Liu, W. 2023. A Theory of Transfer-Based Black-Box Attacks: Explanation and Implications. In *NeurIPS*.

Cohen, J. M.; Rosenfeld, E.; and Kolter, J. Z. 2019. Certified Adversarial Robustness via Randomized Smoothing. In *ICML*, volume 97, 1310–1320.

Devlin, J.; Chang, M.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT*, 4171–4186.

Dvijotham, K. D.; Hayes, J.; Balle, B.; Kolter, J. Z.; Qin, C.; György, A.; Xiao, K.; Gowal, S.; and Kohli, P. 2020. A Framework for robustness Certification of Smoothed Classifiers using F-Divergences. In *ICLR*.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2015. Explaining and Harnessing Adversarial Examples. In *ICLR*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. In *CVPR*, 770–778.

Jeong, J.; Park, S.; Kim, M.; Lee, H.; Kim, D.; and Shin, J. 2021. SmoothMix: Training Confidence-calibrated Smoothed Classifiers for Certified Robustness. In *NeurIPS*, volume 34, 30153–30168.

Jeong, J.; and Shin, J. 2020. Consistency Regularization for Certified Robustness of Smoothed Classifiers. In *NeurIPS*, volume 33, 10558–10570.

Katz, G.; Barrett, C. W.; Dill, D. L.; Julian, K.; and Kochenderfer, M. J. 2017. Reluplex: An Efficient SMT Solver for Verifying Deep Neural Networks. In *CAV*, volume 10426, 97–117.

Krizhevsky, A.; and Hinton, G. 2009. Learning multiple layers of features from tiny images. Technical report, Citeseer.

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

Lécuyer, M.; Atlidakis, V.; Geambasu, R.; Hsu, D.; and Jana, S. 2019. Certified Robustness to Adversarial Examples with Differential Privacy. In *SP*, 656–672.

Lee, G.; Yuan, Y.; Chang, S.; and Jaakkola, T. S. 2019. Tight Certificates of Adversarial Robustness for Randomly Smoothed Classifiers. In *NeurIPS*, 4911–4922.

Li, B.; Chen, C.; Wang, W.; and Carin, L. 2019. Certified Adversarial Robustness with Additive Noise. In *NeurIPS*, 9459–9469.

Li, B.; and Liu, W. 2023. WAT: Improve the Worst-Class Robustness in Adversarial Training. In *AAAI*, 14982–14990.

Li, L.; Qi, X.; Xie, T.; and Li, B. 2020. SoK: Certified Robustness for Deep Neural Networks. *CoRR*, abs/2009.04131.

Li, X.; Zou, X.; and Liu, W. 2022. Defending Against Adversarial Attacks via Neural Dynamic System. In *NeurIPS*.

Ma, X.; Wang, Z.; and Liu, W. 2022. On the Tradeoff Between Robustness and Fairness. In *NeurIPS*.

Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; and Vladu, A. 2018. Towards Deep Learning Models Resistant to Adversarial Attacks. In *ICLR*.

Mao, Y.; Wang, Z.; Liu, W.; Lin, X.; and Hu, W. 2021. BanditMTL: Bandit-based Multi-task Learning for Text Classification. In *ACL*, 5506–5516.

Mao, Y.; Wang, Z.; Liu, W.; Lin, X.; and Hu, W. 2023. Task Variance Regularized Multi-Task Learning. *IEEE Transactions on Knowledge and Data Engineering*, 35(8): 8615–8629.

Mao, Y.; Wang, Z.; Liu, W.; Lin, X.; and Xie, P. 2022. MetaWeighting: Learning to Weight Tasks in Multi-Task Learning. In *Findings of the Association for Computational Linguistics*, 3436–3448.

Metropolis, N.; and Ulam, S. 1949. The Monte Carlo Method. *Journal of the American statistical association*, 44(247): 335–341.

Salman, H.; Li, J.; Razenshteyn, I. P.; Zhang, P.; Zhang, H.; Bubeck, S.; and Yang, G. 2019. Provably Robust Deep Learning via Adversarially Trained Smoothed Classifiers. In *NeurIPS*, 11289–11300.

Shi, L.; and Liu, W. 2023. Adversarial Self-Training Improves Robustness and Generalization for Gradual Domain Adaptation. In *NeurIPS*.

Silver, D.; Schrittwieser, J.; Simonyan, K.; Antonoglou, I.; Huang, A.; Guez, A.; Hubert, T.; Baker, L.; Lai, M.; Bolton, A.; Chen, Y.; Lillicrap, T. P.; Hui, F.; Sifre, L.; van den Driessche, G.; Graepel, T.; and Hassabis, D. 2017. Mastering the game of Go without human knowledge. *Nature*, 550(7676): 354–359.

Sinha, A.; Namkoong, H.; and Duchi, J. C. 2018. Certifying Some Distributional Robustness with Principled Adversarial Training. In *ICLR*.

Tramèr, F.; Carlini, N.; Brendel, W.; and Madry, A. 2020. On Adaptive Attacks to Adversarial Example Defenses. In *NeurIPS*.

Tsipras, D.; Santurkar, S.; Engstrom, L.; Turner, A.; and Madry, A. 2019. Robustness May Be at Odds with Accuracy. In *ICLR*.

Uesato, J.; O'Donoghue, B.; Kohli, P.; and van den Oord, A. 2018. Adversarial Risk and the Dangers of Evaluating Against Weak Attacks. In *ICML*, volume 80, 5032–5041.

Wang, Z.; and Liu, W. 2022. Robustness Verification for Contrastive Learning. In *ICML*, volume 162, 22865–22883.

Wang, Z.; and Liu, W. 2023. RVCL: Evaluating the Robustness of Contrastive Learning via Verification. *Journal of Machine Learning Research*.

Wang, Z.; Pang, T.; Du, C.; Lin, M.; Liu, W.; and Yan, S. 2023. Better Diffusion Models Further Improve Adversarial Training. In *ICML*, volume 202, 36246–36263.

Wong, E.; Schmidt, F. R.; Metzen, J. H.; and Kolter, J. Z. 2018. Scaling provable adversarial defenses. In *NeurIPS*, 8410–8419.

Xu, J.; and Liu, W. 2022. On Robust Multiclass Learnability. In *NeurIPS*.

Xu, J.; and Liu, W. 2023. Characterization of Overfitting in Robust Multiclass Classification. In *NeurIPS*.

Yang, G.; Duan, T.; Hu, J. E.; Salman, H.; Razenshteyn, I. P.; and Li, J. 2020. Randomized Smoothing of All Shapes and Sizes. In *ICML*, volume 119, 10693–10705.

Zhai, R.; Dan, C.; He, D.; Zhang, H.; Gong, B.; Ravikumar, P.; Hsieh, C.; and Wang, L. 2020. MACER: Attack-free and Scalable Robust Training via Maximizing Certified Radius. In *ICLR*.

Zhang, H.; Yu, Y.; Jiao, J.; Xing, E. P.; Ghaoui, L. E.; and Jordan, M. I. 2019. Theoretically Principled Trade-off between Robustness and Accuracy. In *ICML*, volume 97, 7472–7482.

Zhou, Z.; and Liu, W. 2023. Sample Complexity for Distributionally Robust Learning under chi-square divergence. *Journal of Machine Learning Research*, 24: 230:1–230:27.

Zou, X.; and Liu, W. 2023a. Generalization Bounds for Adversarial Contrastive Learning. *Journal of Machine Learning Research*, 24: 114:1–114:54.

Zou, X.; and Liu, W. 2023b. On the Adversarial Robustness of Out-of-distribution Generalization Models. In *NeurIPS*.