Certified Distributional Robustness on Smoothed Classifiers

Jungang Yang ⊙, Liyao Xiang ⊙, Pengzhi Chu ⊙, Xinbing Wang ⊙, and Chenghu Zhou ⊙

Abstract—The robustness of deep neural networks (DNNs) against adversarial example attacks has raised wide attention. For smoothed classifiers, we propose the worst-case adversarial loss over input distributions as a robustness certificate. Compared with previous certificates, our certificate better describes the empirical performance of the smoothed classifiers. By exploiting duality and the smoothness property, we provide an easy-to-compute upper bound as a surrogate for the certificate. We adopt a noisy adversarial learning procedure to minimize the surrogate loss to improve model robustness. We show that our training method provides a theoretically tighter bound over the distributional robust base classifiers. Experiments on a variety of datasets further demonstrate superior robustness performance of our method over the state-of-the-art certified or heuristic methods.

Index Terms—Adversarial learning, certified distributional robustness, randomized smoothing.

I. INTRODUCTION

Deep neural networks (DNNs) have been known to be vulnerable to adversarial example attacks: by feeding the DNN with slightly perturbed inputs, the attack alters the prediction output. The attack can be fatal in performance-critical systems such as autonomous vehicles. A classifier is robust when it can resist such an attack that, as long as the range of the perturbation is not too large (usually invisible by humans), the classifier produces an expected output despite of the specific perturbation. A certifiably robust classifier is one whose prediction at any point \( x \) is verifiably constant within some set around \( x \).

A conventional way to obtain a certifiable robust classifier is to perform randomized smoothing [11, 2, 3, 4]. Assume a base classifier \( f \) tries to map instance \( x_0 \) to corresponding label \( y \). It is found that when fed with instance \( x \) perturbed from \( x_0 \), the smoothed classifier \( g(x) = \mathbb{E}_z [f(x + z)] \) provably returns the same label as \( g(x_0) \) does.

We observe that, even the prediction \( g(x) \) does not alter from \( g(x_0) \) under adversarial perturbations, there is no guarantee that \( g(x_0) \) would return \( y \). It is possible that the adversarially perturbed input has the same label as the original one which is wrongly classified by \( g \). In fact, previous certificates are derived instance-wise, meaning that for one particular instance, there is a distortion bound within which the prediction result does not vary. However, it is unknown how the smoothed classifier performs on the input sample population, which we think is an important robustness indicator since it directly relates to the empirical accuracy of the classifier.

We propose a distributional robustness certificate for smoothed classifiers. We postulate the inputs to the classifier are drawn from a data-generating distribution, and there is a class of distributions around the data-generating distribution. The problem is to seek one distribution that maximizes the loss over a smoothed classifier. It is clear that such a certificate is associated with the empirical accuracy: under the same amount of perturbation, a classifier with a smaller worst-case adversarial loss enjoys higher accuracy over the input population. We prove that the smoothed classifier \( g(\cdot) \) typically has a tighter certificate (a lower loss) than its corresponding base classifier \( f(\cdot) \), suggesting higher robustness against adversarial examples.

To obtain a robust DNN, we minimize the above worst-case adversarial loss over the input distribution. Let \( \ell(\cdot) \) be the loss function and the classifier be parameterized by \( \theta \in \Theta \). The perturbed input is \( s = x + z \sim P \oplus Z \) where \( P \oplus Z \) is the joint distribution of adversarial examples and the Gaussian noise. We aim at minimizing the following certificate:

\[
\minimize_{\theta \in \Theta} \sup_{P \oplus Z} \mathbb{E}_P [\ell(\theta; s)].
\] (1)

However, it still remains problematic how to obtain the above distributional robustness certificate in practice. Therefore, we derive an upper bound for the worst-case adversarial loss of smoothed classifiers and optimize the upper bound as a surrogate loss. We show that the optimal surrogate loss is still inferior to the worst-case adversarial loss of the base classifier, and can be obtained by a noisy adversarial learning procedure. From the training perspective, our method can be considered as augmenting the adversarial examples with Gaussian noise so that instead of minimizing over a limited number of adversarial examples, we minimize over a larger adversarial region where the expected perturbation loss is the worst. As a result, the trained classifier is more robust since it has seen a well-depicted adversarial example distribution.

Manuscript received 20 May 2022; revised 30 January 2023; accepted 21 March 2023. Date of publication 5 April 2023; date of current version 14 March 2024. This work was supported in part by the National Key R&D Program of China under Grants 2021ZD0112801, in part by the NSF China under Grants 62272306, 62032020, 62136006, 42050105, 6202016005, 62061146002, and 61960206002. (Corresponding author: Liyao Xiang.)

Jungang Yang, Liyao Xiang, Pengzhi Chu, and Xinbing Wang are with the Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: yangjungang@sjtu.edu.cn; xiangliyao08@sjtu.edu.cn; pzchu@sjtu.edu.cn; xwang8@sjtu.edu.cn).

Chenghu Zhou is with the Chinese Academy of Sciences, Beijing 100045, China (e-mail: zhouch@lreis.ac.cn).

This article has supplementary downloadable material available at https://doi.org/10.1109/TDSC.2023.3264850, provided by the authors.

Digital Object Identifier 10.1109/TDSC.2023.3264850

1545-5971 © 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
Fig. 1. Illustration of intuitions of ADT and our NAL mechanism. The concentric circles represent the distributions of $\delta$ in [7] and $P, Z$ in our method. We show the adversarial example $x$ which the inner maximization problem finds starting from original sample $x_0$. $x_{0} + \delta_1, x_{0} + \delta_2$ are samples from the distribution surrounding $x_0$, and $x', z'$ are adversarial examples found starting from $x_0 + z_1, x_0 + z_2$. While $x_0 + \delta_1, x_0 + \delta_2$ are adversarial examples found by ADT. Our method takes the neighborhood of $x_0$ into account.

Compared to vanilla adversarial training [5], [6], our approach does not seek one single data point, but rather a neighborhood around the adversarial examples that maximizes the loss in the inner maximization. Compared to adversarial distributional training (ADT) [7], our method models the worst-case changes in data distribution rather than a worst-case adversarial distribution around the natural inputs. To facilitate understanding, we depict Fig. 1 for a comparison of the intuitions behind ADT and our method. We show the difference in seeking the adversarial example from $x_0$ by one step. Starting from $x_0$, ADT (left) looks for distribution parameters which bring the maximal expected loss, and it essentially searches for a $\delta$ direction to march towards maximizing the loss. This is because the highest loss can only be obtained at a single point, and inclusion of any other point will bring down loss. On the contrary, our method seeks a (Gaussian) distribution which has the maximal expected loss under the distribution distance constraint from $x_0$. The loss takes the neighborhood of $x$ into account, not only the point of $x$. More importantly, the distance constraint we apply provides a way to certify the robustness performance. Distinguishing from other previous robustness certificates via smoothed classifiers [1], [3], [4], [8], [9], our method provides a provable guarantee w.r.t. the ground truth input distribution, which better illustrates the robustness of a DNN than the distortion range-based certificate.

Highlights of our contribution are as follows. First, we propose a distributional robustness certificate over noisy inputs, and such a certificate better captures the empirical performance for the smoothed classifiers. Second, we prove the advantage of smoothed classifiers over the base ones from the perspective of certificates, implying a more robust performance for the deployed models. Third, we derive a data-dependent upper bound for the certificate, and minimize it in the training loop. The smoothness property entails the computational tractability of the certificate. We conduct extensive experiments on MNIST, CIFAR-10 and TinyImageNet, comparing with the state-of-the-art adversarial training methods as well as randomized smoothing based methods. The experimental results demonstrate that our method excels in empirical robustness.

II. RELATED WORK

Works proposed to defend against adversarial example attacks can be categorized as follows.

In empirical defences, there is no guarantee how the DNN model would perform against the adversarial examples. Stability training [10], [11] improves model robustness by adding randomized noise to the input during training but shows limited performance enhancement. Adversarial training [5], [12], [13], [14] trains over adversarial examples found at each training step but unfortunately does not guarantee the performance over unseen adversarial inputs. Although without a guarantee, adversarial training has excellent performance in empirical defences against adversarial attacks.

Certified defences are certifiably robust against any adversarial input within an $\ell_p$-norm perturbation range from the original input. Many works use Satisfiability Module Theory (SMT) ([15], [16], [17], [18]) to prove the robustness of neural networks against adversarial attacks. A line of works construct computationally tractable relaxations for computing an upper bound on the worst-case loss. The relaxations include linear programming [19], mixed integer programming [20], semidefinite programming [21], and convex relaxation [22], [23], [24]. And abstract interpretation is also used to verifying neural networks, which can soundly approximate the behavior of neural networks [25], [26]. In addition, [27], [28] consider the boundaries of different samples are non-uniform, so they find the non-uniform boundary of adversarial examples to guarantee the asymmetric robustness of models. However, the above works can only be implemented on small-scale CNN networks, and can not be applied to complex models. [29] also propose a robustness certificate based on a Lagrangian relaxation of the loss function, and is provably robust against adversarial inputs drawn from a distribution centered around the original input distribution. However, it requires the loss function to be smooth and thus prohibits the use of ReLU activation. Similarly, [30] makes a variety of assumptions to the loss function and the data distribution to construct a tractable form for the Wasserstein distributionally robust optimization problem, while our work removes such assumptions and deals with the general case.

Randomized smoothing introduces randomized noise to the neural network, and tries to provide a statistically certified robustness guarantee. The smoothing method does not depend on a specific neural network, or a type of relaxation, but can be generally applied to arbitrary neural networks. The idea of adding randomized noise was first proposed by [4], inspired by the differential privacy protection, and [3] improve the certificate with Rényi divergence. [1] obtain a more relaxed certified robustness bound through the smoothed classifier based on Neyman-Pearson theorem. [31] extend the noise addition mechanism to large-scale parallel algorithms. By extending the randomized noise to the general family of exponential distributions, [2] unify previous approaches to preserve robustness to adversarial attacks. [8] offer adversarial robustness guarantees for $\ell_0$-norm attacks. [32] provides a theoretical certificate for $\ell_0$-norm robustness by a non-additive and deterministic smoothing method. However, the randomized smoothing workflows are questioned.
by [33], to which our work has a similar observation that applying noise augmentation in training does not help with the learning objective. Both [9], [34] employ adversarial training to improve the performance of randomized smoothing. Likewise, our work trains over adversarial data with randomized noise. But we provide a more practical robustness certificate and a training method achieving higher empirical accuracies than theirs.

To better tell the difference between our work and prior ones, we list Table I to compare the loss functions in terms of additive noise for randomization, incorporation of regularization terms, and whether treating inputs as from a data distribution. Our method takes all three components into account.

III. PROPOSED APPROACH

We first define the closeness between distributions, based on which we depict how far the input distribution is perturbed. Under the perturbation constraint, we introduce the robustness certificate for smoothed classifiers. Our main theorem gives a tractable robustness certificate which is easy to be optimized. Following the certificate, we illustrate our algorithm for improving the robustness of the smoothed classifiers.

A. A Distributional Robustness Certificate

Definition 1 (Wasserstein distance). Wasserstein distances notion a closeness between distributions. Let \( \mathcal{X} \subseteq \mathbb{R}^d \), \( A, P \) be a probability space and the transportation cost \( c: \mathcal{X} \times \mathcal{X} \rightarrow [0, \infty) \) be nonnegative, lower semi-continuous, and \( c(x, x) = 0 \). \( P \) and \( Q \) are two probability measures supported on \( \mathcal{X} \). Let \( \Pi(P, Q) \) denotes the collection of all measures on \( \mathcal{X} \times \mathcal{X} \) with marginals \( P \) and \( Q \) on the first and second factors respectively, i.e., it holds that \( \pi(A, \mathcal{X}) = P(A) \) and \( \pi(\mathcal{X}, A) = Q(A), \forall A \in A \) and \( \pi \in \Pi(P, Q) \). The Wasserstein distance between \( P \) and \( Q \) is

\[
W_c(P, Q) := \inf_{\pi \in \Pi(P, Q)} \mathbb{E}_\pi [c(x, y)].
\]

(2)

For example, the \( \ell_2 \)-norm \( c(x, x_0) = \|x - x_0\|_2^2 \) satisfies the aforementioned conditions.

**Distributional robustness for smoothed classifiers:** Assume the original input \( x_0 \) is drawn from the distribution \( P_0 \), and the perturbed input \( z \) is drawn from the distribution \( P \). Since the perturbed input should be visually indistinguishable from the original one, we define the robustness region for the smoothed classifier as \( \mathcal{P} = \{ P : W_c(P \oplus Z, P_0) \leq \rho, P \in \mathcal{P}(\mathcal{X}) \} \) where \( \rho > 0 \). Instead of regarding the noise as a part of the smoothed classifier, we let \( z = x + z \) be a noisy input coming from the distribution \( P \oplus Z \). We use \( p_x, p_z \) to denote the probability density function of \( x, z \). The probability density function of \( s \) can be written as:

\[
p_x(s) = \int_{\mathbb{R}^d} p_z(t)p_z(s - t) \, dt.
\]

(3)

Since the noise \( z \in \mathbb{R}^d \), we need to set \( \mathcal{X} = \mathbb{R}^d \) to admit \( x + z \in \mathcal{X} \) as [1], [4], [9] do. Within such a region, we evaluate the robustness as a worst-case population loss over noisy inputs:

\[
\sup_{P: W_c(P \oplus Z, P_0) \leq \rho} \mathbb{E}_{P \oplus Z}[\ell(\theta; s)] \leq \sup_{P: W_c(P \oplus Z, P_0) \leq \rho} \mathbb{E}_P[\ell(\theta; x')].
\]

(4)

\( x' \sim P' \) is also the adversarial example in order to distinguish \( P \) on the left-hand. The proof is straightforward. The right-hand side of (4) indicates the worst-case loss over all adversarial distributions which are at most \( \rho \) away from \( P_0 \). It is intuitive that the one with Gaussian noise is a special case of the adversarial distributions and hence its worst-case loss should be no larger. The theorem illustrates that within a given perturbation range, the smoothed classifier potentially provides a lower adversarial loss over the input distribution and therefore higher robustness. However, such a robustness metric is impossible to measure in practice as we cannot depict \( P \) precisely. Even if \( P \) can be acquired, it can be a non-convex region which renders the constrained optimization objective intractable. Hence we resort to the Lagrangian relaxation of the problem to derive an upper bound for it.

B. A Surrogate Loss

As the main theorem of this work, we provide an upper bound for the worst-case population loss for any level of robustness \( \rho \). We further show that for small enough \( \rho \), the upper bound is tractable and easy to optimize.

**Theorem 2.** Let \( \ell : \Theta \times \mathcal{X} \rightarrow \mathbb{R} \) and transportation cost function \( c: \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}_+ \) be continuous. Let \( x_0 \) be an input drawn from the input distribution \( P_0 \), \( x \) be the adversarial example which follows the distribution \( P \) and \( z \sim Z = N(0, \sigma^2 I) \) be the additive noise of the same shape as \( x \). Let \( \phi_\gamma(\theta; x_0) = \sup_{x \in \mathcal{X}} \mathbb{E}_Z[\ell(\theta; x + z) - \gamma c(x + z, x_0)] \) be the robust surrogate. For any \( \gamma, \rho > 0 \) and \( \sigma \), we have

\[
\sup_{P: W_c(P \oplus Z, P_0) \leq \rho} \mathbb{E}_{P \oplus Z}[\ell(\theta; s)] \leq \gamma \rho + \mathbb{E}_{P_0}[\phi_\gamma(\theta; x_0)].
\]

(5)
Proof scheme: To find the upper bound of \( \sup_{P:W_c(P)\leq \rho} E_{P\otimes Z}[\ell(\theta; s)] \), we first apply a Lagrangian form to transform the problem into its duality with dual variable \( \gamma \). We rewrite the expectation in integral form, and apply the definition of Wasserstein distance. The integral form is converted into \( \phi_\gamma(\theta; x_0) \) by separating the random variable \( z \) from \( x \), taking the expectation of \( z \), and maximizing over \( x \). We complete the proof by the definition of \( \sup \) and \( \inf \).

Proof. We express the worst-case loss in its dual form with dual variable \( \gamma \). By the weak dual property, we have

\[
\begin{align*}
\sup_{P:W_c(P)\leq \rho} E_{P\otimes Z}[\ell(\theta; s)] & \leq \\
\inf_{\gamma \geq 0} \sup_{P:W_c(P)\leq \rho} \{ E_{P\otimes Z}[\ell(\theta; s)] - \gamma W_c(P \otimes Z, P_0) + \gamma \rho \},
\end{align*}
\]

(6)

the left hand-side of which can be rewritten in integral form:

\[
\begin{align*}
\inf_{\gamma \geq 0} \sup_{P:W_c(P)\leq \rho} & \{ E_{P\otimes Z}[\ell(\theta; x + z)] - \gamma W_c(P \otimes Z, P_0) + \gamma \rho \} \\
& = \sup_{\gamma \geq 0} \inf_{P:W_c(P)\leq \rho} \{ \gamma W_c(P \otimes Z, P_0) + \gamma \rho \} \\
& \times \left\{ \int [\ell(\theta; x + z) - \gamma W_c(P \otimes Z, P_0) + \gamma \rho] d\pi(x) \right\},
\end{align*}
\]

(7)

Note that for any \( \pi \in \Pi(P \otimes Z, P_0) \), we have the statement that \( \int f(s) dP \otimes Z = \int f(s) d\pi(s, x_0) \). And by the definition of Wasserstein distance, we have

\[
\begin{align*}
\inf_{\gamma \geq 0} \sup_{P:W_c(P)\leq \rho} & \times \left\{ \int [\ell(\theta; s) - \gamma W_c(P \otimes Z, P_0) + \gamma \rho] d\pi(s, x_0) \right\} \\
& = \sup_{\gamma \geq 0} \inf_{P:W_c(P)\leq \rho} \left\{ \int \ell(\theta; s) d\pi(s, x_0) \right\} \\
& - \gamma \inf_{\pi \in \Pi(P \otimes Z, P_0)} \int c(s, x_0) d\pi(s, x_0) + \gamma \rho \\
& = \sup_{\gamma \geq 0} \inf_{P:W_c(P)\leq \rho} \left\{ \int \ell(\theta; s) - \gamma c(s, x_0) d\pi(s, x_0) + \gamma \rho \right\},
\end{align*}
\]

(8)

By the independence between \( z \) and \( x, x_0 \), one would obtain

\[
\begin{align*}
\int [\ell(\theta; s) - \gamma c(s, x_0)] d\pi(s, x_0) & = \int \int [\ell(\theta; x + z) - \gamma c(x + z, x_0)] d\pi(x, x_0) \\
& = \int \int [\ell(\theta; x + z) - \gamma c(x + z, x_0)] d\pi(x, x_0)
\end{align*}
\]

(9)

By taking the maximum over \( x \),

\[
\begin{align*}
\int \int [\ell(\theta; x + z) - \gamma c(x + z, x_0)] d\pi(x, x_0) & = \int \int [\ell(\theta; x + z) - \gamma c(x + z, x_0)] d\pi(x, x_0) \\
& \leq \int \sup_x \{ E_Z[\ell(\theta; x + z) - \gamma c(x + z, x_0)] \} d\pi(x, x_0).
\end{align*}
\]

(10)

Fixing \( x \) to be value that maximizes the expression to be integrated, \( x \) in the formula is fixed, so we only need to integrate \( d\pi(x, x_0) \) on \( X \). So we can get:

\[
\begin{align*}
\int \sup_x \{ E_Z[\ell(\theta; x + z) - \gamma c(x + z, x_0)] \} d\pi(x, x_0) & = \int \sup_x \{ E_Z[\ell(\theta; x + z) - \gamma c(x + z, x_0)] \} dP_0(x_0) \\
& = E_{P_0} \sup_x E_Z[\ell(\theta; x + z) - \gamma c(x + z, x_0)].
\end{align*}
\]

(11)

Because the distribution of \( z \) is definite and \( z \) is independent of \( x \), and supremum of \( P \otimes Z \) is replaced by the supremum of \( x \). Therefore, (8) can be written as

\[
\begin{align*}
\inf_{\gamma \geq 0} \sup_{P:W_c(P)\leq \rho} & \left\{ \sup_{\pi \in \Pi(P \otimes Z, P_0)} \int [\ell(\theta; s) - \gamma c(s, x_0)] d\pi(s, x_0) + \gamma \rho \right\} \\
& \leq \inf_{\gamma \geq 0} \sup_{P:W_c(P)\leq \rho} \sup_{\pi \in \Pi(P \otimes Z, P_0)} \left\{ E_{P_0} \sup_x E_Z[\ell(\theta; x + z) - \gamma c(x + z, x_0)] + \gamma \rho \right\}.
\end{align*}
\]

(12)

By plugging the above into (6), we could get

\[
\begin{align*}
\sup_{P:W_c(P)\leq \rho} E_{P\otimes Z}[\ell(\theta; s)] & \leq \inf_{\gamma \geq 0} \left\{ E_{P_0} \sup_x E_Z[\ell(\theta; x + z) - \gamma c(x + z, x_0)] + \gamma \rho \right\} \\
& = \inf_{\gamma \geq 0} \left\{ E_{P_0} [\phi_\gamma(\theta; x_0)] + \gamma \rho \right\}.
\end{align*}
\]

(13)

for any given \( \gamma \geq 0 \), which completes the proof.

It is notable that the right-hand side takes the expectation over \( P_0 \) and \( Z \) respectively. Given a particular input \( x_0 \sim P_0 \), we seek an adversarial example that maximizes the expected loss over the additive noise. Typically, \( P_0 \) is impossible to obtain and thus we use an empirical distribution, such as the training data distribution, to approximate \( P_0 \) in practice.

Since Theorem 2 provides an upper bound for the worst-case population loss, it offers a principled adversarial training approach which minimizes the upper bound instead of the actual loss, i.e.,

\[
\begin{align*}
\text{minimize } & E_{P_0}[\phi_\gamma(\theta; x_0)],
\end{align*}
\]

(14)
In the following, we show the above loss function has a tractable form for arbitrary neural networks, due to a smoothed loss function. Hence Theorem 2 provides a tractable robustness certificate depending on the data.

Properties of the Smoothed Classifier: We show the optimization objective of (14) is easy to compute for any neural network, particularly for the non-smooth ones with ReLU activation layers. More importantly, the smoothness of the classifier enables the adversarial training procedure to converge as we want by using the common optimization techniques such as stochastic gradient descent. The smoothness of the loss function comes from the smoothed classifier with randomized noise. Specifically,

Lemma 1. Assume \( \ell : \Theta \times \mathcal{X} \to [0, M] \) is a bounded loss function. The loss function on the smoothed classifier can be expressed as \( \hat{\ell}(\theta; x) := \mathbb{E}_z[\ell(\theta; x + z)] \), \( z \sim \mathcal{N}(0, \sigma^2 I) \). Then we have \( \hat{\ell} \) is \( \frac{2M}{\sigma^2} \)-smooth w.r.t. \( \ell_2 \)-norm, i.e., \( \| \nabla_x \hat{\ell}(\theta; x) - \nabla_{x'} \hat{\ell}(\theta; x') \|_2 \leq \frac{2M}{\sigma^2} \| x - x' \|_2 \).

We present the proof in Appendix A.1, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/10.1109/TDSC.2023.3264850. It mainly takes advantage of the randomized noise which has a smoothing effect on the loss function. For DNNs with non-smooth layers, the smoothed classifier makes it up and turns the loss function to a smoothed one, which contributes as an important property to the strong concavity of \( \mathbb{E}_z[\ell(\theta; x + z) - \gamma c(x + z, x_0)] \) and therefore ensures the tractability of the robustness certificate. Please refer to supplementary material for more properties.

C. Noisy Adversarial Learning Algorithm

Problem (14) provides an explicit way to improve the robustness of a smoothed classifier parameterized by \( \theta \). We correspondingly design a noisy adversarial learning algorithm to obtain a classifier whose robustness can be guaranteed. In the algorithm, we use the empirical distribution to replace the ideal input distribution \( P_0 \), and sample \( z \) \( r \) times to substitute the expectation with the sample average. Assuming we have a total of \( n \) training instances \( x_0^i, \forall i \in [n] \), and sample \( z_{ij} \sim \mathcal{N}(0, \sigma^2 I) \) for the \( i \)th instance for \( r \) times, the objective is:

\[
\min_{\theta \in \Theta} \frac{1}{nr} \sum_{i=1}^{n} \sup_{x \in \mathcal{X}} \sum_{j=1}^{r} \left[ \ell(\theta; x + z_{ij}) - \gamma c(x + z_{ij}, x_0^i) \right].
\]

(16)

The detail of the algorithm is illustrated in Algorithm 1. In the inner maximization step (line 3-6), we adopt the projected gradient descent (PGD) [5, 12] to approximate the maximizer according to the convention. The hyperparameters include the number of iterations \( K \) and the learning rate \( \eta_1 \). Within each iteration, we sample the Gaussian noise \( z \) \( r \) times, given which we compute an average perturbation direction for each update. The more noise samples, the closer the averaging result is to the expected value, which is at the sacrifice of higher computation expense. Similarly, a larger number of \( K \) indicates stronger adversarial attacks and higher model robustness, but also incurs higher computation complexity. Hence choosing appropriate values of \( r \) and \( K \) is important in practice.

After training is done, we obtain the classifier parameter \( \theta \). In the inference phase, we sample a number of \( z \sim \mathcal{N}(0, \sigma^2 I) \) to add to the testing instance. The noisy testing examples are fed to the classifier to get the prediction outputs.

An Alternative View: We provide an alternative intuition for our algorithm. Assume that the input dataset contains \( n \) records and \( X_0 \) is a sample of the distribution \( P^g_0 \) since each input \( x_0 \sim P_0 \). Correspondingly, let the adversarial sample set be \( X \) which is drawn from the worst-case adversarial distribution \( P^\star \), and \( P^\star = \arg \sup_{\gamma \in \mathcal{P}} \mathbb{E}_{P}[\ell(\theta; x)] \). A significant drawback of the vanilla adversarial training is that, the loss value at each training sample could be quite different, resulting in high instability in the model robustness. Instead of training over a limited number of inputs, we wish to train over a larger input set to reduce the instability. Ideally, we train over \( m \) samples of \( P^g \) rather than one: \( X^{(1)}, \ldots, X^{(m)} \sim P^g \). Obviously, we do not have so much data, so we adopt a sampled mean \( M = \sum_{i=1}^{m} X^{(i)}/m \) for training. It is intuitively more stable to train over \( M \) than over \( X \). To acquire \( M \), we apply Central Limit Theorem for an estimation: \( m \to \infty, M \sim \mathcal{N}(\mu, \sigma^2 I) \), where \( \mu \) and \( \sigma^2 \) are the mean and variance of the distribution \( P^g \). By using the worst-case adversarial examples \( X \) to approximate \( \mu \) and \( \sigma^2 \) to estimate \( \tilde{\mathcal{L}}^g_m \), our algorithm trains over \( M \sim \mathcal{N}(X, \sigma^2 I) \). It means that each input is added the Gaussian noise \( z \sim Z = \mathcal{N}(0, \sigma^2 I) \) before being fed to the classifier. The trick is called randomized smoothing in [1, 4, 9]. Hence, we show an alternative explanation for why randomized smoothing could enhance the robustness of the classifiers.

Convergence: An important property associated with the smoothed classifier is the strong concavity of the robust surrogate loss, which is the key to the convergence proof. As long as the loss \( \hat{\ell} \) is smooth on the parameter space \( \Theta \), NAL has a convergence rate \( O(1/\sqrt{T}) \), similar to [29], but NAL does not need to replace the non-smooth layer ReLU with Sigmoid or ELU to guarantee robustness.
We start with the required assumptions, which roughly quantify the robustness we provide.

**Assumption 1.** The loss $\hat{\ell}: \Theta \times \mathcal{X} \rightarrow [0, M]$ satisfies the Lipschitz-smoothness conditions

$$
\| \nabla_{\theta} \hat{\ell}(\theta; x) - \nabla_{\theta} \hat{\ell}(\theta'; x) \|_2 \leq L_{\theta\theta} \| \theta - \theta' \|, \\
\| \nabla_{x} \hat{\ell}(\theta; x) - \nabla_{x} \hat{\ell}(\theta; x') \|_2 \leq L_{xx} \| x - x' \|, \\
\| \nabla_{\theta} \hat{\ell}(\theta; x) - \nabla_{\theta} \hat{\ell}(\theta; x') \|_2 \leq L_{\theta x} \| x - x' \|, \\
\| \nabla_{x} \hat{\ell}(\theta; x) - \nabla_{x} \hat{\ell}(\theta; x') \|_2 \leq L_{x x} \| x - x' \|.
$$

(17)

Let $\| \cdot \|$ be the dual norm to $\| \cdot \|$ on $\Theta$ and $\mathcal{X}$.

In order to present the proof of convergence, we first need to prove that the function $\mathbb{E}_{Z}[\ell(\theta; x + z) - \gamma c(x + z, x_{0})]$ is strongly concave.

**Proposition 1.** For any $c: \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R} \cup \{\infty\}$ 1-strongly convex in its first argument, and $\ell: \theta \times x \rightarrow \mathbb{E}_{Z}[\ell(\theta; x + z)]$ being $\frac{2M}{\sigma^2}$-smooth, the function $\mathbb{E}_{Z}[\ell(\theta; x + z) - \gamma c(x + z, x_{0})]$ is strongly concave in $x$ for any $\gamma \geq \frac{2M}{\sigma^2}$.

**Proof.** Since $\hat{\ell}$ is $\frac{2M}{\sigma^2}$-smooth and $c$ is 1-strongly convex in its first argument, we have

$$
\nabla_{x}^2 \mathbb{E}_{Z}[\ell(\theta; x + z) - \gamma c(x + z, x_{0})] \leq \frac{2M}{\sigma^2} I,
$$

(18)

and

$$
\nabla_{x}^2 \mathbb{E}_{Z}[\ell(\theta; x + z) - \gamma c(x + z, x_{0})] \leq \frac{2M}{\sigma^2} \gamma I.
$$

(19)

Therefore we have

$$
\nabla_{x}^2 \mathbb{E}_{Z}[\ell(\theta; x + z) - \gamma c(x + z, x_{0})] \leq \left( \frac{2M}{\sigma^2} - \gamma \right) I.
$$

(20)

Hence the strong concavity is proved for $\gamma \geq \frac{2M}{\sigma^2}$. \hfill \Box

**Lemma 2.** Let $f : \Theta \times \mathcal{X} \rightarrow \mathbb{R}$ be differentiable and $\lambda$-strongly convex in $x$ with respect to the norm $\| \cdot \|$, and define $\bar{f}(\theta) = \sup_{x \in \mathcal{X}} f(\theta, x)$. Let $g_0(\theta, x) = \nabla_{\theta} \bar{f}(\theta, x)$ and $g_x(\theta, x) = \nabla_{x} \bar{f}(\theta, x)$, and assume $g_0$ and $g_x$ satisfy Assumption 1 with $\bar{f}(\theta, x)$ replaced with $f(\theta, x)$. Then $\bar{f}$ is differentiable, and letting $x^*(\theta) = \arg\max_{x \in \mathcal{X}} f(\theta, x)$, we have $\nabla \bar{f}(\theta) = g_0(\theta, x^*(\theta))$.

Moreover,

$$
\| x^*(\theta_1) - x^*(\theta_2) \| \leq \frac{L_{xx}}{\lambda} \| \theta_1 - \theta_2 \|,
$$

and

$$
\| \nabla \bar{f}(\theta_1) - \nabla \bar{f}(\theta_2) \| \leq \left( L_{\theta \theta} + \frac{L_{\theta x} L_{xx}}{\gamma L_{xx}} \right) \| \theta_1 - \theta_2 \|,
$$

(21)

which imply that $\bar{f}(\theta)$ is $(L_{\theta \theta} + \frac{L_{\theta x} L_{xx}}{\gamma L_{xx}})$-smooth.

The proof is referred to Appendix A.2, available in the online supplemental material.

Fix $x_0 \in \mathcal{X}$ and focus on the $\ell_2$-norm case where $c(x, x_0)$ satisfies 1-strongly convex with $\| \cdot \|_2$. Noting that $f(\theta, x) := \mathbb{E}_{Z}[\ell(\theta; x + z) - \gamma c(x + z, x_0)]$ is $\gamma$ strongly convex, then has $L_{\theta \theta} + \frac{L_{\theta x} L_{xx}}{\gamma L_{xx}}$-Lipschitz gradients, and

$$
\nabla_{\theta} \phi_{\gamma}(\theta; x_{0}) = \nabla_{\theta} \mathbb{E}_{Z}[\ell(\theta; x + z) - \gamma c(x + z, x_{0})],
$$

(22)

where $x^*(x_0, \theta) = \arg\max_{x \in \mathcal{X}} \mathbb{E}_{Z}[\ell(\theta; x + z) - \gamma c(x + z, x_{0})]$, noting that we perform gradient steps with

$$
g^t = \nabla \bar{f}(\theta^t; x^t, x_{0})
$$

for $x'$ an $\epsilon$-approximate maximizer of $f(\theta; x, x_0)$ in $x$ (it means that $f(\theta; x', x_0) - f(\theta; x, x_0) \leq \epsilon$, and $\theta^{t+1} = \theta_t - \eta \gamma^t$).

We assume $\eta \leq \frac{1}{2L_{\theta \theta}}$ in the rest of the proof. According to Lemma 2, we could know that $F$ is $L_{\theta \theta}$-smooth. By a Taylor expansion using the $L_{\theta \theta}$-smoothness of the objective function, we have

$$
F(\theta^{t+1}) \leq F(\theta^t) + \langle \nabla F(\theta^t), \theta^{t+1} - \theta^t \rangle + \frac{L_{\theta \theta}}{2} \| \theta^{t+1} - \theta^t \|_2^2
$$

$$
= F(\theta^t) - \eta \left( 1 - \frac{1}{2} L_{\theta \theta} \eta \right) \| \nabla F(\theta^t) \|_2^2
$$

$$
+ \frac{L_{\theta \theta} \eta^2}{2} \| g^t - \nabla F(\theta^t) \|_2^2
$$

$$
+ \eta \left( 1 - L_{\theta \theta} \eta \right) \langle \nabla F(\theta^t), \nabla F(\theta^t) - g^t \rangle.
$$

(24)
Recalling the definition $\phi_\gamma(\theta; x_0) = E_x \sup_{z \in S} \{ f(\theta; x + z) - \gamma c(x + z, x_0) \}$, we define the potentially biased errors $\delta^t = \nabla_\theta \phi_\gamma(\theta; x_0) - \theta^t$. Hence
\[
F(\theta^{t+1}) \leq F(\theta^t) - \eta_2 \left( 1 - \frac{1}{2} L_\phi \eta_2 \right) \| \nabla F(\theta^t) \|^2_2 + \eta_2 \left( 1 - L_\phi \eta_2 \right) \langle \nabla F(\theta^t), \nabla F(\theta^t) - \nabla_\theta \phi_\gamma(\theta; x_0) \rangle + L_\phi \eta_2^2 / 2 \| \nabla_\theta \phi_\gamma(\theta; x_0) + \delta^t - \nabla F(\theta^t) \|^2_2 - \eta_2 \left( 1 - L_\phi \eta_2 \right) \langle \nabla F(\theta^t), \delta^t \rangle.
\]
(25)

Using $+\langle a, b \rangle \leq \frac{1}{2} \| a \|^2 + \frac{1}{2} \| b \|^2$ and $\| a \| \leq \| b \|$ in the preceding display, we get
\[
F(\theta^{t+1}) \leq F(\theta^t) + L_\phi \eta_2^2 \| \nabla F(\theta^t) - \nabla_\theta \phi_\gamma(\theta; x_0) \|^2_2 - \eta_2 \frac{L_\phi}{2} \| \nabla F(\theta^t) \|^2_2 + \frac{\eta_2 (1 + L_\phi \eta_2)}{2} \| \delta^t \|^2_2 + \eta_2 \left( 1 - L_\phi \eta_2 \right) \langle \nabla F(\theta^t), \nabla F(\theta^t) - \nabla_\theta \phi_\gamma(\theta; x_0) \rangle.
\]
(26)

Using $E[\nabla_\theta \phi_\gamma(\theta^t; z^t) | \theta^t] = \nabla F(\theta^t)$ and the assumption in Theorem 3, we derive
\[
E \left[ F(\theta^{t+1}) - F(\theta^t) | \theta^t \right] \leq -\eta_2 \frac{L_\phi}{2} \| \nabla F(\theta^t) \|^2_2 + \frac{\eta_2 \| \nabla_\theta \phi_\gamma(\theta^t; x_0) + \delta^t - \nabla F(\theta^t) \|^2_2}{2} + L_\phi \eta_2^2 \tau^2.
\]
(27)

Letting $x^* = \arg\max_x f(\theta^t; x, x_0)$, note that the error $\delta^t$ satisfies
\[
\| \delta^t \|^2_2 = \| \nabla_\theta \phi_\gamma(\theta^t; x_0) - \nabla_\theta f(\theta^t; x^*, x_0) \|^2_2
\]
\[
= \| \nabla_\theta \epsilon(\theta^t; x^*, x_0) \|^2_2
\]
\[
\leq L_{\theta,x} \| x^* - x \|^2_2 \frac{L_\phi^2 \tau}{\lambda}.
\]
(28)

where the final inequality uses the $\lambda = \gamma - L_{xx}$ strong-concavity of $x \mapsto f(\theta^t; x, x_0)$. For shorthand, let $\tilde{\epsilon} = \frac{L_{\theta,x}^2 \tau}{\lambda} \epsilon$. We have
\[
E \left[ F(\theta^{t+1}) - F(\theta^t) | \theta^t \right] \leq -\eta_2 \frac{L_\phi}{2} \| \nabla F(\theta^t) \|^2_2 + \eta_2 \tilde{\epsilon} + L_\phi \eta_2^2 \tau^2.
\]
(29)

where we use the fact that $\eta_2 \leq \frac{1}{L_\phi \eta_2}$. For a fixed stepsize $\eta_2$, taking the total expectation yields
\[
E \left[ \| \nabla F(\theta^t) \|^2_2 \right] - 2\tilde{\epsilon} \leq \frac{2}{\eta_2} E \left[ F(\theta^t) - F(\theta^{t+1}) \right] + 2L_\phi \eta_2^2 \tau^2.
\]
(30)

Since $F(\theta^t) - F(\theta^T) \leq F(\theta^0) - \inf_{\theta} F(\theta) \leq \Delta F$ by assumption, we sum over $t$ to obtain
\[
\frac{1}{T} \sum_{t=0}^{T-1} E \left[ \| \nabla F(\theta^t) \|^2_2 \right] - 2\tilde{\epsilon} \leq \frac{2}{\eta_2} E \left[ F(\theta^0) - F(\theta^T) \right] + 2L_\phi \eta_2^2 \tau^2.
\]
(31)

Applying $\eta_2 = \sqrt{\frac{\Delta F}{2L_\phi \tau^2}}$, we could get the result that
\[
\frac{1}{T} \sum_{t=0}^{T-1} E \left[ \| \nabla F(\theta^t) \|^2_2 \right] - 2\tilde{\epsilon} \leq 4T \sqrt{\frac{\Delta F}{L_\phi \tau^2}}.
\]
(32)

It is straightforward from the conclusion that a larger number of iterations facilitates convergence. For the variance $\sigma^2$ of the Gaussian noise, we observe that its main influence is to $L_{xx} = M \frac{\sigma^2}{\tau}$. Therefore, the larger $\sigma$, the smaller $L_{xx}$, resulting in a smaller $L_\phi$ which means the robustness certificate is tighter. Since there is a natural tradeoff between adversarial robustness and accuracy [13], the enhancement in adversarial robustness leads to a decrease in accuracy. We will further elaborate on the points in the experiments.

IV. COMPARISON WITH OTHER CERTIFICATES

We compare our work with the state-of-the-art robustness definitions and certificates in this section.

A. Adversarial Training

In Theorem 1, we show that the distributional robustness certificate improves over smoothed classifiers. Now we further demonstrate that our robust surrogate loss is still inferior to the worst-case adversarial loss of the base classifier.

Corollary 1. Under the same denotations and conditions as Theorem 2, we have
\[
\inf_{\gamma \geq 0} \{ \gamma \rho + E_{P_0}[\phi_\gamma(\theta; x_0)] \} \leq \sup_{P \leq W_\epsilon(\rho \gamma ; P_0)} E_{P}[\ell(\theta; s)],
\]
(30)

where $x' \sim P'$ is also the adversarial example and the right-hand-side is the same of that in Theorem 1.

We present the proof here. Proof. By (13) we could get
\[
\sup_{P \leq W_\epsilon(\rho \gamma ; P_0)} E_{P}[\ell(\theta; s)] \leq \inf_{\gamma \geq 0} \{ \gamma \rho + E_{P_0}[\phi_\gamma(\theta; x_0)] \},
\]
(31)

where
\[
E_{P_0}[\phi_\gamma(\theta; x_0)] = E_{P_0} \left\{ \sup_{x \in X} E_{Z}[\ell(\theta; x + z) - \gamma c(x + z, x_0)] \right\}.
\]
(32)

Since $\ell(\theta; x + z) - \gamma c(x + z, x_0)$ is strongly concave for $x + z$ in [29], by Jensen Inequality we have for any fixed $x$,
\[
E_{Z}[\ell(\theta; x + z) - \gamma c(x + z, x_0)] \leq \ell(\theta; x) - \gamma c(x, x_0).
\]
(33)

Hence, the following inequality holds
\[
E_{P_0}[\phi_\gamma(\theta; x_0)] \leq E_{P_0} \sup_{x \in X} [\ell(\theta; x) - \gamma c(x, x_0)].
\]
(34)
attacks. Undefended means a naturally trained model. Solid lines represent $R > \ell$ and others guarantee the to and . Actually, we found the are within a given range with probability $\sigma$ and can also be found in Theorem 7 of [29] attack radius. $(\epsilon)$ sup respectively. For example, the results in $p$ and is $\leq 2$ $\epsilon$. (34) Testing accuracies under $=0$ $g$ is the inverse of the standard Gaussian CDF, $=2 \gamma \rho$ is $\leq 0$ is applications, not a single instance. (triangle) degrade significantly compared with STN/training with noise (diamond/circle). The result is a piece of evidence that a classifier almost cannot defend adversarial attacks when trained without but tested with additive noise (solid line with triangles). Therefore, we conclude the smoothed classifier can only improve robustness only if the base classifier is robust. We consider robustness refers to the ability of a DNN to classify adversarial examples into the correct classes, and such an ability should be evaluated on the population of adversarial examples, not a single instance.

V. EXPERIMENTS

Metrics: To evaluate the certified accuracies for the smoothed classifiers, we perform 100000 tests on each test example to calculate $p_A$ and $\overline{p_B}$ respectively, and then use (35) to calculate the robust radius $R$. We compute certified accuracy by counting the proportion of examples whose robust radius $R$ is larger than $\ell_2$ attack radius. $R > \ell_2$ attack radius means the example is robust. The same setting of [1] is adopted to evaluate the certified accuracy. We also evaluate the empirical accuracies for different methods by launching the PGD attack [5], [12] following the convention of [3], [13], [29], etc. We set the number of iterations in PGD attack as $K_{PGD} = 20$, 100 respectively and the learning rate $\eta = 2\epsilon / K_{PGD}$ where $\epsilon$ is $\ell_2$ attack radius.

Baselines, Datasets and Models. Testing accuracies under different levels of adversarial attacks are chosen as the metric. We compare the empirical performance of NAL with representative baselines including: WRM [29], SmoothAdv [9], STN [3], smoothing [1], PGD [5] and TRADES [13]. Since WRM...
Fig. 3. (a), (b), and (c) We compare the certified accuracies of NAL with SmoothAdv, Smoothing and STN on MNIST, CIFAR-10 and Tiny ImageNet at $\gamma = 1.5$ and the corresponding $\varepsilon$. On MNIST and Tiny ImageNet, NAL shows a close certified accuracy to SmoothAdv while superior to other baselines overall. On CIFAR-10, the performance of NAL is close to STN, slightly stronger than SmoothAdv and Smoothing. (d) We compare NAL with SmoothAdv, TRADES and STN on Tiny ImageNet at $\gamma = 1.5$ and the corresponding $\varepsilon$. NAL shows a better performance than all baselines, except the natural accuracy ($\ell_2$ attack radius $= 0$) of STN.

| Dataset         | $\gamma$ | $\varepsilon$ |
|-----------------|----------|---------------|
| MNIST           | 0.5/1.5  | (0.84, 0.34, 0.21) |
| CIFAR-10 (R)    | 0.5/1.5  | (1.53, 0.92, 0.40) |
| CIFAR-10 (V)    | 0.5/1.5  | (1.23, 0.57, 0.28) |
| Tiny ImageNet   | 0.5/1.5  | 0.93          |

TABLE II
HYPERPARAMETERS AND PERTURBATION RANGE ON DIFFERENT DATASETS. CIFAR-10 (R) REPRESENTS CIFAR-10, RESNET-18 AND CIFAR-10 (V) REPRESENTS CIFAR-10, VGG-16

Training Hyperparameters. Table II gives the training hyperparameters in NAL and the batch size is chosen as 128. The hyperparameters used in baselines are supplied in Table III.

A. Results
Due to space constraints, we only show partial results. Please find the complete results in supplementary material.

Fig. 4. Trade-off between robustness and natural accuracy on CIFAR-10, ResNet-18 with $\sigma = 0.1$ and different $\varepsilon$s. The dotted line shows the general trend of baselines with each $\varepsilon$. The same color represents the same setting with corresponding $\gamma$ and $\varepsilon$, under which NAL has the best performance overall.

We compute $\rho$ as the expected transportation cost between the generated adversarial examples and the original inputs over the training set:

$$\varepsilon^2 = \rho(\theta) = \mathbb{E}_{P_0} \mathbb{E}_Z [c(x + z, x_0)].$$

(36)

And $\varepsilon$ can be computed accordingly. The corresponding values of $\gamma$ and $\varepsilon$ are given in Table II as well.

A. Results
Due to space constraints, we only show partial results. Please find the complete results in supplementary material.

TABLE IV
TESTING ACCURACIES FOR ReLU MODEL AND ELU MODEL ON CIFAR-10, RESNET-18 WITH NAL AND FIXED $\sigma = 0.1$

empirical accuracies of different methods are presented in Figs. 3(d), 4, and Table V. For WRM-related comparison, all experiments are conducted on ELU-modified DNNs to ensure smoothness by the request from [29]. NAL exceeds WRM for every $\gamma$ over all the datasets including MNIST, CIFAR-10 and Tiny ImageNet. Empirical accuracies of other baselines are presented in Table V. NAL outperforms all others in the sum of the robust and natural accuracies while STN, Smoothing and PGD enjoy relatively higher natural accuracy. This may be explained by the inherent trade-off between natural accuracy
and robustness [13]. In addition, TRADES and SmoothAdv have better robustness performance than STN, Smoothing and PGD. And the sum of robust and natural accuracies of TRADES and SmoothAdv are higher. Therefore, the performance of TRADES and SmoothAdv will be better than STN, Smoothing and PGD. Since the classification difficulty of the MNIST dataset is relatively simple, the results of most of the mechanisms are not significantly different, and the difference between CIFAR-10 and Tiny ImageNet will be more obvious. Since we test robust accuracy against PGD attacks, the adversarial training with PGD has the worst performance among all. The adversarial examples generated by PGD attacks would yield a maximal loss in the inner maximization problem in PGD adversarial training.

To show NAL indeed has a better tradeoff, we depict the robustness-accuracy with different \( \varepsilon \) in Fig. 4 on CIFAR 10, ResNet-18. Most of the baselines exhibit some kind of tradeoff except that NAL has superior performance above all. We made the corresponding results of all experiments into coordinates “(natural accuracy, robust accuracy)”. The shapes of different legends in the figure represent different defense methods, and different colors represent the experimental results under different parameter (\( \varepsilon \) or \( \gamma \)) settings. Under different parameter settings, we use dashed lines of different colors to mark the straight line of the trade-off between natural accuracy and robust accuracy. It can be seen that all the baselines are near the dashed line. The results show that these baseline methods are different performances under a close trade-off. The difference is that TRADES and SmoothAdv are located on the left side of the dashed line, so robust accuracy is higher, and STN, Smoothing and PGD are located on the right side of the dashed line, so natural accuracy is higher. In addition, as the \( \varepsilon \) increases, we find that the dotted line moves to the lower left, which means that the trade-off performance of the model decreases as the \( \varepsilon \) increases. In the figure, we can clearly find that the star pattern representing the STN method is above the dotted line of the same color, and the gap is obvious. It can be seen that the STN method not only has better robust accuracy, but also has better trade-off performance. We present more result in Appendix B.2, available in the online supplemental material.

ReLU and ELU Models. To show the advantage of our approach applying to ReLU models, we compare ResNet-18 with ReLU and ELU on CIFAR-10 for NAL. From Fig. 2(c), throughout the training process, the loss of the ReLU model is smaller than that of the ELU model, and ReLU model presents faster convergence. The robustness performance of both models is presented in Table IV. It is clear that in the testing phase, the ReLU model also obtains a better performance. And the only one exception is that when \( \ell_2 \) radius = 0, the accuracy of two models is very close. Hence ReLU models have better performance than ELU models in general, demonstrating our approach avoids accuracy degradation caused by ELU networks due to the requirement of smooth property. Compare With WRM on ELU-Based Models. Fig. 5 shows the comparison between NAL and WRM on the ELU-based models under the PGD-20 attack over different databases. And NAL outperforms WRM over all the datasets and different models with different \( \gamma \)s. We can see that even under the same ELU-based models, the NAL method still present a better performance than the WRM method. MNIST looks relatively close due to its simple task, and obvious differences can be seen on CIFAR-10 and Tiny ImageNet. Since the NAL mechanism has better performance on the ReLU model, the direct gap with WRM will further increase.

Sample Number and PGD Iterations. We also study the impact of the noise sample number \( r \) and PGD iteration \( K \) to the model robustness with CIFAR-10, ResNet-18 as an example. The results in Table VI show that while the model performance enhances with \( K \), it does not necessarily increase with a larger \( r \). For a combined consideration of computation overhead and accuracy, we choose \( K = 4 \), \( r = 4 \) by default, which is likely to deliver sufficiently good performance.

Noise Level. We vary the value of \( \sigma \) in the experiments to find out their impact. By the results in Table VII, we observe \( \sigma = 0.1 \) yields the best performance on CIFAR-10, considering all levels of adversarial attacks. For SmoothAdv and STN, \( \sigma = 0.1 \) also has good performance. Although STN has the highest natural accuracy when \( \sigma = 0.05 \), the robustness of this model is too low when under PGD attack. Hence, we choose \( \sigma = 0.1 \) by default in the following. Here we only show the data of \( \gamma = 1.5 \) and \( (K, r) = (4, 4) \). For more complete data, please refer to the supplementary materials.

Size of the Datasets. We also study the impact of dataset size to robustness and accuracy on MNIST, CNN as an example. Table VIII shows the results for varying training data sizes. The overall impact of dataset sizes is mild in our testing range. Nevertheless, both natural accuracy and robust accuracy improved with a larger dataset. Although the highest accuracies are not obtained at the largest dataset, the results are sufficient to support that a larger dataset leads to more accurate estimation of the data distribution, thereby yielding superior performance.
Fig. 5. NAL outperforms WRM on MNIST, CNN, CIFAR-10, VGG-16, CIFAR-10, ResNet-18 and Tiny ImageNet, ResNet-18. All the experiments are trained on ELU models under different $\gamma$s. For the same $\gamma$, NAL exceeds WRM.

### TABLE VII

| $\ell_2$ attack radius | 0   | 0.25 | 0.5  | 0.75 | 1.25 | 1.5  | 1.75 |
|-------------------------|-----|------|------|------|------|------|------|
| NAL $\sigma = 0.05$    | 0.8579 | 0.7809 | 0.6761 | 0.5549 | 0.4262 | 0.2916 | 0.1888 | 0.1329 |
| NAL $\sigma = 0.1$     | 0.8522 | 0.8155 | 0.7684 | 0.7140 | 0.6466 | 0.5684 | 0.4829 | 0.3909 |
| NAL $\sigma = 0.2$     | 0.8307 | 0.7781 | 0.7213 | 0.6498 | 0.5644 | 0.4785 | 0.3837 | 0.2999 |
| SmoothAdv $\sigma = 0.05$ | 0.7643 | 0.7086 | 0.6378 | 0.5644 | 0.4841 | 0.4050 | 0.3297 | 0.2602 |
| SmoothAdv $\sigma = 0.1$ | 0.8066 | 0.7264 | 0.6281 | 0.5376 | 0.4399 | 0.3467 | 0.2700 | 0.2010 |
| SmoothAdv $\sigma = 0.2$ | 0.7411 | 0.6758 | 0.6079 | 0.5327 | 0.4689 | 0.4005 | 0.3350 | 0.2736 |
| STN $\sigma = 0.05$    | 0.8988 | 0.7347 | 0.4834 | 0.2594 | 0.1167 | 0.0466 | 0.0155 | 0.0063 |
| STN $\sigma = 0.1$     | 0.8669 | 0.7609 | 0.6164 | 0.4116 | 0.2847 | 0.1678 | 0.0927 | 0.0444 |
| STN $\sigma = 0.2$     | 0.8000 | 0.7060 | 0.5867 | 0.4695 | 0.3523 | 0.2472 | 0.1708 | 0.1123 |

### TABLE VIII

The impact of the dataset size on MNIST, CNN. We set the $\gamma = 1.5$, $\epsilon = 0.34$, $\sigma = 0.1$ and the robust accuracy is obtained under PGD-20 attack. The best performance at the same noise level is in bold.

| MNIST-CNN | Num of dataset |
|-----------|----------------|
| Natural   | 10000 20000 30000 40000 50000 60000 |
| Robust    | 98.61% 98.94% 99.04% 99.27% 99.14% 99.18% |

**Certificate:** To better understand how close the upper bound is to the true distributional risk, we plot our certificate $\gamma_{\rho} + \mathbb{E}_{P_{\text{test}}} [\phi(\theta; x_0)]$ against any level of robustness $\rho$, and the out-of-sample (test) worst-case performance $\sup_{S \in \mathbb{F}} \mathbb{E}_S [\ell(\theta; s)]$ for NAL (Fig. 2(b)). Since the worst-case loss is almost impossible to evaluate, we solve its Lagrangian relaxation for different values of $\gamma$: for each chosen $\gamma$, we compute the average distance to adversarial examples in the test set as $\hat{\rho}_{\text{test}}(\theta)$, $\mathbb{E}_{P_{\text{test}}} [\mathbb{E}_Z [c(x_{\star} + z, x_0)]]$ where $P_{\text{test}}$ is the test data distribution and $x_{\star} = \arg \max_x \mathbb{E}_Z [\ell(\theta; x + z) - \gamma c(x + z, x_0)]$ is the adversarial perturbation of $x_0$. The worst-case loss is given by $(\hat{\rho}_{\text{test}}(\theta), \mathbb{E}_{P_{\text{test}}} [\mathbb{E}_Z [\ell(\theta; x_{\star} + z)]]$. We also observe that, $\hat{\rho}_{\text{test}}(\theta)$ tends to increase with a higher noise level. Hence we need to keep the noise at an appropriate level to make our certificate tractable.

**Certified Accuracies of NAL, SmoothAdv, Smoothing and STN on MNIST, CIFAR-10 and Tiny ImageNet are given in Fig. 3(a), (b), and (c). We found that NAL has a close performance to SmoothAdv but is superior to others on MNIST and Tiny ImageNet. On CIFAR-10, NAL has a close performance to STN but is also superior to others. Smoothing does not fit well to these datasets, as most $R$ values are the same, due to a lack of regularization terms. Although the performance of NAL on Tiny ImageNet is inferior to SmoothAdv when $\ell_2$ attack radius in the range of (0.2, 0.4), it is better than SmoothAdv in the range of (0, 0.2), therefore we consider that the performance of these two methods is close. We present more result in Appendix B.1, available in the online supplemental material.**

**Transferrability.** Now we evaluate the robustness of defenses on CIFAR-10 under black-box attacks to perform a thorough evaluation following the guidelines in [36]. We evaluate transfer-based black-box attacks using PGD-20. The results are presented in Fig. 2(d). The vertical axis represents the source model where the attack is generated, and the horizontal axis represents the target model. The diagonal denotes the accuracy of the target model under white-box attacks, and non-diagonal elements are accuracies of the target models under black-box attacks. Obviously, NAL has a higher accuracy in defending white-box attacks and the accuracies under black-box attacks are moderately high. Overall, we observe NAL has a better defence performance (column-wise) whereas the attack generated by NAL is weaker (row-wise), indicating the NAL model is suitable for deployment.
VI. CONCLUSION

Our work views the robustness of a smoothed classifier from a different perspective, i.e., the worst-case population loss over the input distribution. We provide a tractable upper bound (certificate) for the loss and devise a noisy adversarial learning approach to obtain a tight certificate. Compared with previous works, our certificate is practically meaningful and offers superior empirical robustness performance.

ACKNOWLEDGMENTS

Authors would like to appreciate the Student Innovation Center of SJTU for providing GPUs.

REFERENCES

[1] J. Cohen, E. Rosenfeld, and Z. Kolter, “Certified adversarial robustness via randomized smoothing,” in Proc. Int. Conf. Mach. Learn., 2019, pp. 1310–1320.

[2] R. Pinot et al., “Theoretical evidence for adversarial robustness through randomization,” in Proc. Adv. Neural Inf. Process. Syst., 2019, pp. 11838–11848.

[3] B. Li, C. Chen, W. Wang, and L. Carin, “Certified adversarial robustness with additive noise,” in Proc. Adv. Neural Inf. Process. Syst., 2019, pp. 9464–9474.

[4] M. Lecuyer, V. Atlidakis, R. Geambasu, D. Hsu, and S. Jana, “Certified robustness to adversarial examples with differential privacy,” in Proc. IEEE Symp. Secur. Privacy, 2019, pp. 656–672.

[5] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, “Towards deep learning models resistant to adversarial attacks,” in Proc. Int. Conf. Learn. Representations, 2018. [Online]. Available: https://openreview.net/forum?id=rJzIBfZAb

[6] N. Phan et al., “Heterogeneous Gaussian mechanism: Preserving differential privacy in deep learning with provable robustness,” in Proc. 26th Int. Joint Conf. Artif. Intell., 2017, pp. 4753–4759.

[7] Z. Deng, Y. Dong, T. Pang, H. Su, and J. Zhu, “Adversarial distributional training for robust deep learning,” Adv. Neural Inf. Process. Syst., vol. 33, pp. 8270–8283, 2020.

[8] G.-H. Lee, Y. Yuan, S. Chang, and T. Jaakkola, “ Tight certificates of adversarial robustness for randomly smoothed classifiers,” in Proc. Adv. Neural Inf. Process. Syst., 2019, pp. 4910–4921.

[9] H. Salman et al., “Provably robust deep learning via adversarially trained smooth classifiers,” in Proc. Adv. Neural Inf. Process. Syst., 2019, pp. 11292–11303.

[10] S. Zheng, Y. Song, T. Leung, and I. Goodfellow, “Improving the robustness of deep neural networks via stability training,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, pp. 4480–4488.

[11] V. Zantedeschi, M.-J. Nicolae, and A. Rawat, “Efficient defenses against adversarial attacks,” in Proc. 10th ACM Workshop Artif. Intell. Secur., 2017, pp. 39–49.

[12] A. Kurakin, I. Goodfellow, and S. Bengio, “Adversarial machine learning at scale,” in Proc. Int. Conf. Learn. Representations, 2017. [Online]. Available: https://openreview.net/forum?id=ByNtFB4Kg

[13] H. Zhang, Y. Yu, J. Jiao, E. Xing, L. El Ghaoui, and M. I. Jordan, “Theoretically principled trade-off between robustness and accuracy,” in Proc. 36th Int. Conf. Mach. Learn., Long Beach, California, USA, vol. 97, Jun. 9–15, 2019, pp. 7472–7482.

[14] Y. Wang, D. Zou, J. Yi, J. Bailey, X. Ma, and Q. Gu, “Improving adversarial robustness requires reweighting misclassified examples,” in Proc. Int. Conf. Learn. Representations, 2019. [Online]. Available: https://openreview.net/forum?id=rkOg8EFpW

[15] X. Huang, M. Kwiatkowska, S. Wang, and M. Wu, “Safety verification of deep neural networks,” in Proc. Int. Conf. Comput. Aided Verification, 2017, pp. 3–29.

[16] G. Katz, C. Barrett, D. L. Dill, K. Julian, and M. J. Kochenderfer, “Re-lupex: An efficient SMT solver for verifying deep neural networks,” in Proc. Int. Conf. Comput. Aided Verification, 2017, pp. 97–117.

[17] R. Ehlers, “ Formal verification of piece-wise linear feed-forward neural networks,” in Proc. Int. Symp. Automated Technol. Verification Anal., 2017, pp. 269–286.

[18] N. Carlini, G. Katz, C. Barrett, and D. L. Dill, “Ground-truth adversarial examples,” 2017, arXiv abs/1709.10207.

[19] E. Wong and Z. Kolter, “Provable defenses against adversarial examples via the convex outer adversarial polytope,” in Proc. Int. Conf. Mach. Learn., 2018, pp. 5286–5295.

[20] V. Tjeng, K. Y. Xiao, and R. Tedrake, “Evaluating robustness of neural networks with mixed integer programming,” in Proc. Int. Conf. Learn. Representations, 2018. [Online]. Available: https://openreview.net/forum?id=HyGldIqRt

[21] A. Raghunathan, J. Steinhardt, and P. S. Liang, “ Semidefinite relaxations for certifying robustness to adversarial examples,” in Proc. Adv. Neural Inf. Process. Syst., 2018, pp. 10877–10887.

[22] H. Namkoong and J. C. Duchi, “Variance-based regularization with convex objectives,” in Proc. Adv. Neural Inf. Process. Syst., 2017, pp. 2971–2980.

[23] E. Wong, F. Schmidt, J. H. Metzen, and J. Z. Kolter, “Scalable provable adversarial defenses,” in Proc. Adv. Neural Inf. Process. Syst., 2018, pp. 8400–8409.

[24] H. Salman, G. Yang, H. Zhang, C.-J. Hsieh, and P. Zhang, “A convex relaxation barrier to tight robustness verification of neural networks,” in Proc. Adv. Neural Inf. Process. Syst., 2019, pp. 9835–9846.

[25] T. Gehr, M. Mirman, D. Drachsler-Cohen, P. Tsankov, S. Chaudhuri, and M. Vechev, “AI2: Safety and robustness certification of neural networks with abstract interpretation,” in Proc. IEEE Symp. Secur. Privacy, 2018, pp. 3–18.

[26] G. Singh, T. Gehr, M. Mirman, M. Püschel, and M. Vechev, “Fast and effective robustness certification,” in Proc. Adv. Neural Inf. Process. Syst., vol. 31, 2018, pp. 10825–10836.

[27] C. Liu, R. Tomioka, and V. Cherkasov, “On certifying non-uniform bounds against adversarial attacks,” in Proc. Int. Conf. Mach. Learn., 2019, pp. 4072–4081.

[28] C. Li et al., “Towards certifying the asymmetric robustness for neural networks: Quantification and applications,” IEEE Trans. Dependable Secure Comput., vol. 19, no. 6, pp. 3987–4001, Nov./Dec. 2022.

[29] A. Sinha, H. Namkoong, and J. Duchi, “Certifying some distributional robustness with principled adversarial training,” in Proc. Int. Conf. Learn. Representations, 2018. [Online]. Available: https://openreview.net/forum?id=bK6kPqZ-A

[30] D. Kuhn, P. M. Esfahani, V. A. Nguyen, and S. Shafieezadeh-Abadeh, “Wasserstein distributionally robust optimization: Theory and applications in machine learning,” in Proc. Operations Res. Manage. Sci. Age Analytics, 2019, pp. 130–166.

[31] N. Phan, M. T. Thai, H. Hu, R. Jin, T. Sun, and D. Dou, “Scalable differential privacy with certified robustness in adversarial learning,” in Proc. Int. Conf. Mach. Learn. vol. 119, 2019, pp. 7683–7694.

[32] A. Levine and S. Feizi, “Improved, deterministic smoothing for L1 certified robustness,” 2021, arXiv:2103.10834.

[33] J. Mohapatra, C.-Y. Ko, L. Weng, P.-Y. Chen, S. Liu, and L. Daniel, “Hidden cost of randomized smoothing,” in Proc. Int. Conf. Artif. Intell. Statist., 2021, pp. 4033–4041.

[34] J. Jia, X. Cao, B. Wang, and N. Z. Gong, “Certified robustness for top-k predictions against adversarial perturbations via randomized smoothing,” in Proc. Int. Conf. Learn. Representations, 2019. [Online]. Available: https://openreview.net/forum?id=Bk6w6W6FwVf

[35] A. Juditsky, A. Nemirovski, and C. Tavaul, “Solving variational inequalities with stochastic mirror-prox algorithm,” Stochastic Syst., vol. 1, no. 1, pp. 59–88, 2011.

[36] N. Carlini et al., “On evaluating adversarial robustness,” 2019, arXiv:1902.06705.
Pengzhi Chu received the BEng degree in electrical and information engineering from Xi’an Jiao Tong University, Xi’an, Shaanxi, China, in 2011, and the MEng degree, in 2018. He is currently a research assistant with Shanghai Jiao Tong University and a Director assistant of Student Innovation Center. His research interests include autonomous driving, model compression and network deployment.

Xinbing Wang received the BS degree (with hons.) from the Department of Automation, Shanghai Jiao-tong University, Shanghai, China, in 1998, and the MS degree from the Department of Computer Science and Technology, Tsinghua University, Beijing, China, in 2001, and received the PhD degree, major in the Department of electrical and Computer Engineering, minor in the Department of Mathematics, North Carolina State University, Raleigh, in 2006. Currently, he is a professor in the Department of Electronic Engineering, Shanghai Jiaotong University, Shanghai, China. He has been an associate editor for IEEE/ACM Transactions on Networking and IEEE Transactions on Mobile Computing, and the member of the Technical Program Committees of several conferences including ACM MobiCom 2012, ACM MobiHoc 2012-2014, IEEE INFOCOM 2009-2017.

Chenghu Zhou received the BS degree in geography from Nanjing University, Nanjing, China, in 1984, and the MS and PhD degrees in geographic information system from the Chinese Academy of Sciences (CAS), Beijing, China, in 1987 and 1992, respectively. He is currently an academician with CAS, where he is also a research professor with the Institute of Geographical Sciences and Natural Resources Research, and a professor with the School of Geography and Ocean Science, Nanjing University. His research interests include spatial and temporal data mining, geographic modeling, hydrology and water resources, and geographic information systems and remote sensing applications.