Proceedings of Machine Learning Research vol 242:1–16, 2024

Mixing Classifiers to Alleviate the Accuracy-Robustness Trade-Off

Yatong Bai YATONG_BAI@BERKELEY.EDU
Brendon G. Anderson BGANDERSON@BERKELEY.EDU
Somayeh Sojoudi SOJOUDI@BERKELEY.EDU
University of California, Berkeley

Abstract

Deep neural classifiers have recently found tremendous success in data-driven control systems. However, existing models suffer from a trade-off between accuracy and adversarial robustness. This limitation must be overcome in the control of safety-critical systems that require both high performance and rigorous robustness guarantees. In this work, we develop classifiers that simultaneously inherit high robustness from robust models and high accuracy from standard models. Specifically, we propose a theoretically motivated formulation that mixes the output probabilities of a standard neural network and a robust neural network. Both base classifiers are pre-trained, and thus our method does not require additional training. Our numerical experiments verify that the mixed classifier noticeably improves the accuracy-robustness trade-off and identify the confidence property of the robust base classifier as the key leverage of this more benign trade-off. Our theoretical results prove that under mild assumptions, when the robustness of the robust base model is certifiable, no alteration or attack within a closed-form ℓ_p radius on an input can result in misclassification of the mixed classifier.[1]

Keywords: Adversarial Robustness, Image Classification, Computer Vision, Model Ensemble

1. Introduction

In recent years, high-performance machine learning models have been employed in various control settings, including reinforcement learning for dynamic systems with uncertainty (Levine et al., 2016; Sutton and Barto, 2018) and autonomous driving (Bojarski et al., 2016; Wu et al., 2017). However, models such as neural networks have been shown to be vulnerable to adversarial attacks, which are imperceptibly small input data alterations maliciously designed to cause failure (Szegedy et al., 2014; Nguyen et al., 2015; Huang et al., 2017; Eykholt et al., 2018; Liu et al., 2019). This vulnerability makes such models unreliable for safety-critical control where guaranteeing robustness is necessary. In response, "adversarial training (AT)" (Kurakin et al., 2017; Goodfellow et al., 2015; Bai et al., 2022a,b; Zheng et al., 2020; Zhang et al., 2019) have been studied to alleviate the susceptibility. AT builds robust neural networks by training on adversarially attacked data.

A parallel line of work focuses on mathematically certified robustness (Anderson et al., 2020; Ma and Sojoudi, 2021; Anderson and Sojoudi, 2022a). Among these methods, "randomized smoothing (RS)" is a particularly popular one that seeks to achieve certified robustness by processing intentionally corrupted data at inference time (Cohen et al., 2019; Li et al., 2019; Pfrommer et al., 2023), and has recently been applied to robustify reinforcement learning-based control strategies (Kumar et al., 2022; Wu et al., 2022). The recent work (Anderson and Sojoudi, 2022b) has shown that "locally biased smoothing," which robustifies the model locally based on the input test datum, outperforms the traditional RS with fixed smoothing noise. However, Anderson and Sojoudi (2022b) only focus on binary classification problems, significantly limiting the applications. Moreover, Anderson and

[1] Supplementary materials at bai-yt.github.io/files/publications/MixedClassiL4DCSupp.pdf.

Sojoudi (2022b) rely on the robustness of a K-nearest-neighbor (K-NN) classifier, which suffers from a lack of representation power when applied to harder problems and becomes a bottleneck.

While some works have shown that there exists a fundamental trade-off between accuracy and robustness (Tsipras et al., 2019; Zhang et al., 2019), recent research has argued that it should be possible to simultaneously achieve robustness and accuracy on benchmark datasets (Yang et al., 2020). To this end, variants of AT that improve the accuracy-robustness trade-off have been proposed, including TRADES (Zhang et al., 2019), Interpolated Adversarial Training (Lamb et al., 2019), and many others (Raghunathan et al., 2020; Zhang and Wang, 2019; Tramèr et al., 2018; Balaji et al., 2019). However, even with these improvements, degraded clean accuracy is often an inevitable price of achieving robustness. Moreover, standard non-robust models often achieve enormous performance gains by pre-training on larger datasets, whereas the effect of pre-training on robust classifiers is less understood and may be less prominent (Chen et al., 2020; Fan et al., 2021).

This work makes a theoretically disciplined step towards robustifying models without sacrificing clean accuracy. Specifically, we build upon locally biased smoothing and replace its underlying K-NN classifier with a robust neural network that can be obtained via various existing methods. We then modify how the standard base model (a highly accurate but possibly non-robust neural network) and the robust base model are "mixed" accordingly. The resulting formulation, to be introduced in Section 3, is a convex combination of the output probabilities from the two base classifiers. We prove that, when the robust network has a bounded Lipschitz constant or is built via RS, the mixed classifier also has a closed-form certified robust radius. More importantly, our method achieves an empirical robustness level close to that of the robust base model while approaching the standard base model's clean accuracy. This desirable behavior significantly improves the accuracy-robustness trade-off, especially for tasks where standard models noticeably outperform robust models on clean data.

Note that we do not make any assumptions about how the standard and robust base models are obtained (can be AT, RS, or others), nor do we assume the adversarial attack type and budget. Thus, our mixed classification scheme can take advantage of pre-training on large datasets via the standard base classifier and benefit from ever-improving robust training methods via the robust base classifier.

2. Background and related works

2.1. Notations

The ℓ_p norm is denoted by $\|\cdot\|_p$, while $\|\cdot\|_{p*}$ denotes its dual norm. The matrix I_d denotes the identity matrix in $\mathbb{R}^{d\times d}$. For a scalar a, $\operatorname{sgn}(a) \in \{-1, 0, 1\}$ denotes its sign. For a natural number c, the set $[c]$ is defined as $\{1, 2, \dots, c\}$. For an event A, the indicator function $\mathbb{I}(A)$ evaluates to 1 if A takes place and 0 otherwise. The notation $\mathbb{P}_{X\sim\mathcal{S}}[A(X)]$ denotes the probability for an event $A(X)$ to occur, where X is a random variable drawn from the distribution $\mathcal{S}$. The normal distribution on $\mathbb{R}^d$ with mean $\overline{x}$ and covariance Σ is written as $\mathcal{N}(\overline{x}, \Sigma)$. We denote the cumulative distribution function of $\mathcal{N}(0, 1)$ on $\mathbb{R}$ by Φ and write its inverse function as Φ^{-1}.

Consider a model $g : \mathbb{R}^d \to \mathbb{R}^c$, whose components are $g_i : \mathbb{R}^d \to \mathbb{R},\ i \in [c]$, where d is the dimension of the input and c is the number of classes. In this paper, we assume that $g(\cdot)$ does not have the desired level of robustness, and refer to it as a "standard model", as opposed to a "robust model" which we denote as $h(\cdot)$. We consider ℓ_p norm-bounded attacks on differentiable neural networks. A classifier $f : \mathbb{R}^d \to [c]$, defined as $f(x) = \arg\max_{i\in[c]} g_i(x)$, is considered robust against adversarial attacks at an input datum $x \in \mathbb{R}^d$ if it assigns the same class to all perturbed inputs $x + \delta$ such that $\|\delta\|_p \leq \epsilon$, where $\epsilon \geq 0$ is the attack radius.

2.2. Related Adversarial Attacks and Defenses

The fast gradient sign method (FGSM) and projected gradient descent (PGD) attacks based on differentiating the cross-entropy loss are highly effective and have been considered the most standard attacks for evaluating robust models (Madry et al., 2018; Goodfellow et al., 2015). To exploit the structures of the defense methods, adaptive attacks have also been introduced (Tramèr et al., 2020).

On the defense side, while AT (Madry et al., 2018) and TRADES (Zhang et al., 2019) have seen enormous success, such methods are often limited by a significantly larger amount of required training data (Schmidt et al., 2018) and a decrease in generalization capability. Initiatives that construct more effective training data via data augmentation (Rebuffi et al., 2021; Gowal et al., 2021) and generative models (Sehwag et al., 2022) have successfully produced more robust models. Improved versions of AT (Jia et al., 2022; Shafahi et al., 2019) have also been proposed.

Previous initiatives that aim to enhance the accuracy-robustness trade-off include using alternative attacks during training (Pang et al., 2022), appending early-exit side branches to a single network (Hu et al., 2020), and applying AT for regularization (Zheng et al., 2021). Moreover, ensemble-based defenses, such as random ensemble (Liu et al., 2018) and diverse ensemble (Pang et al., 2019; Alam et al., 2022), have been proposed. In comparison, this work considers two separate classifiers and uses their synergy to improve the accuracy-robustness trade-off, achieving higher performances.

2.3. Locally Biased Smoothing

Randomized smoothing, popularized by (Cohen et al., 2019), achieves robustness at inference time by replacing $f(x) = \arg\max_{i \in [c]} g_i(x)$ with a smoothed classifier $\widetilde{f}(x) = \arg\max_{i \in [c]} \mathbb{E}_{\xi \sim \mathcal{S}} \left[g_i(x + \xi) \right]$, where $\mathcal{S}$ is a smoothing distribution. A common choice for $\mathcal{S}$ is a Gaussian distribution.

Anderson and Sojoudi (2022b) have recently argued that data-invariant RS does not always achieve robustness. They have shown that in the binary classification setting, RS with an unbiased distribution is suboptimal, and an optimal smoothing procedure shifts the input point in the direction of its true class. Since the true class is generally unavailable, a "direction oracle" is used as a surrogate. This "locally biased smoothing" method is no longer randomized and outperforms traditional data-blind RS. The locally biased smoothed classifier, denoted $h^\gamma \colon \mathbb{R}^d \to \mathbb{R}$, is obtained via the deterministic calculation $h^\gamma(x) = g(x) + \gamma h(x) \|\nabla g(x)\|_{p*}$, where $h(x) \in \{-1, 1\}$ is the direction oracle and $\gamma \geq 0$ is a trade-off parameter. The direction oracle should come from an inherently robust classifier (which is often less accurate). In (Anderson and Sojoudi, 2022b), this direction oracle is chosen to be a one-nearest-neighbor classifier.

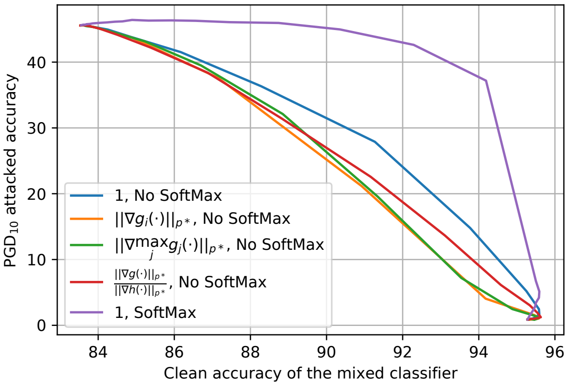
3. Using a Robust Neural Network as the Smoothing Oracle

Locally biased smoothing was designed for binary classification, restricting its practicality. Here, we first extend it to the multi-class setting by treating the output of each class, denoted as $h_i^\gamma(x)$, independently, giving rise to:

$$h^\gamma_{\text{smo1},i}(x) := g_i(x) + \gamma h_i(x) \|\nabla g_i(x)\|_{p*}, \quad i \in [c]. \tag{1}$$

Note that if $\|\nabla g_i(x)\|_{p*}$ is large for some class i, then $h^\gamma_{\text{smo1},i}(x)$ can be large for class i even if both $g_i(x)$ and $h_i(x)$ are small, leading to incorrect predictions. To remove the effect of the gradient magnitude difference across the classes, we propose a normalized formulation as follows:

$$h^\gamma_{\text{smo2},i}(x) := \frac{g_i(x) + \gamma h_i(x) \|\nabla g_i(x)\|_{p*}}{1 + \gamma \|\nabla g_i(x)\|_{p*}}, \quad i \in [c]. \tag{2}$$



- "No Softmax" represents Option 1, i.e., use the logits for $g(\cdot)$ and $h(\cdot)$.
- "Softmax" represents Option 2, i.e., use the probabilities for $g(\cdot)$ and $h(\cdot)$.
- With the best formulation, high clean accuracy can be achieved with very little sacrifice on robustness.

Figure 1: Comparing the "attacked accuracy – clean accuracy" curves for various options for $R_i(x)$.

The parameter γ adjusts between clean accuracy and robustness. It holds that $h^{\gamma}_{\text{smo2},i}(x) \equiv g_i(x)$ when $\gamma = 0$, and $h^{\gamma}_{\text{smo2},i}(x) \to h_i(x)$ when $\gamma \to \infty$ for all x and all i.

With the mixing procedure generalized to the multi-class setting, we now discuss the choice of the smoothing oracle $h_i(\cdot)$. While K-NN classifiers are relatively robust and can be used as the oracle, their representation power is too weak. On the CIFAR-10 image classification task (Krizhevsky, 2012), K-NN only achieves around 35% accuracy on clean test data. In contrast, an adversarially trained ResNet can reach 50% accuracy on attacked test data (Madry et al., 2018). This lackluster performance of K-NN becomes a significant bottleneck in the accuracy-robustness trade-off of the mixed classifier. To this end, we replace the K-NN model with a robust neural network. The robustness of this network can be achieved via various methods, including AT, TRADES, and RS.

Further scrutinizing (2) leads to the question of whether $\|\nabla g_i(x)\|_{p*}$ is the best choice for adjusting the mixture of $g(\cdot)$ and $h(\cdot)$. This gradient magnitude term is a result of Anderson and Sojoudi (2022b)'s assumption that $h(x) \in \{-1, 1\}$. Here, we no longer have this assumption. Instead, we assume both $g(\cdot)$ and $h(\cdot)$ to be differentiable. Thus, we generalize the formulation to

$$h^{\gamma}_{\text{smo3},i}(x) := \frac{g_i(x) + \gamma R_i(x) h_i(x)}{1 + \gamma R_i(x)}, \quad i \in [c], \tag{3}$$

where $R_i(x)$ is an extra scalar term that can potentially depend on both $\nabla g_i(x)$ and $\nabla h_i(x)$ to determine the "trustworthiness" of the base classifiers. Here, we empirically compare four options for $R_i(x)$, namely, 1, $\|\nabla g_i(x)\|_{p*}$, $\|\nabla \max_j g_j(x)\|_{p*}$, and $\frac{\|\nabla g_i(x)\|_{p*}}{\|\nabla h_i(x)\|_{p*}}$.

Another design question is whether $g(\cdot)$ and $h(\cdot)$ should be the pre-softmax logits or the post-softmax probabilities. Note that since most attack methods are designed based on logits, the output of the mixed classifier should be logits rather than probabilities to avoid gradient masking, an undesirable phenomenon that makes it hard to evaluate the robustness properly. Thus, we have the following two options that make the mixed model compatible with existing gradient-based attacks:

1. Use the logits for both base classifiers, $g(\cdot)$ and $h(\cdot)$.
2. Use the probabilities for both base classifiers, and then convert the mixed probabilities back to logits. The required "inverse-softmax" operator is simply the natural logarithm.

Figure 1 visualizes the accuracy-robustness trade-off achieved by mixing logits or probabilities with different $R_i(x)$ options. Here, the base classifiers are a pair of standard and adversarially trained

ResNet-18s. This "clean accuracy versus PGD$_{10}$-attacked accuracy" plot concludes that $R_i(x) = 1$ gives the best accuracy-robustness trade-off, and $g(\cdot)$ and $h(\cdot)$ should be probabilities. Appendix A in the supplementary materials confirms this selection by repeating Figure 1 with alternative model architectures, different robust base classifier training methods, and various attack budgets.

Our selection of $R_i(x) = 1$ differs from $R_i(x) = \|g_i(x)\|_{p*}$ used in (Anderson and Sojoudi, 2022b). Intuitively, Anderson and Sojoudi (2022b) used linear classifiers to motivate estimating the base models' trustworthiness with their gradient magnitudes. When the base classifiers are highly nonlinear neural networks as in our case, while a base classifier's local Lipschitzness correlates with its robustness, its gradient magnitude is not always a good local Lipschitzness estimator. Additionally, Section 3.1 offers theoretical intuitions for selecting mixing probabilities over mixing logits.

With these design choices implemented, the formulation (3) can be re-parameterized as

$$h_i^\alpha(x) := \log\big((1-\alpha)g_i(x) + \alpha h_i(x)\big), \quad i \in [c], \tag{4}$$

where $\alpha = \frac{\gamma}{1+\gamma} \in [0,1]$. We take $h^\alpha(\cdot)$ in (4), which is a convex combination of base classifier probabilities, as our proposed mixed classifier. Note that (4) calculates the mixed classifier logits, acting as a drop-in replacement for existing models which usually produce logits. Removing the logarithm recovers the output probabilities without changing the predicted class.

3.1. Theoretical Certified Robust Radius

In this section, we derive certified robust radii for the mixed classifier $h^\alpha(\cdot)$ introduced in (4), given in terms of the robustness properties of $h(\cdot)$ and the mixing parameter α. The results ensure that despite being more sophisticated than a single model, $h^\alpha(\cdot)$ cannot be easily conquered, even if an adversary attempts to adapt its attack methods to its structure. Such guarantees are of paramount importance for reliable deployment in safety-critical control applications.

Noticing that the base model probabilities satisfy $0 \le g_i(\cdot) \le 1$ and $0 \le h_i(\cdot) \le 1$ for all i, we introduce the following generalized and tightened notion of certified robustness.

Definition 1 *Consider an arbitrary input $x \in \mathbb{R}^d$ and let $y = \arg\max_i h_i(x)$, $\mu \in [0,1]$, and $r \ge 0$. Then, $h(\cdot)$ is said to be* certifiably robust at x with margin μ and radius r *if $h_y(x+\delta) \ge h_i(x+\delta) + \mu$ for all $i \ne y$ and all $\delta \in \mathbb{R}^d$ such that $\|\delta\|_p \le r$.*

Lemma 2 *Let $x \in \mathbb{R}^d$ and $r \ge 0$. If it holds that $\alpha \in [\frac{1}{2}, 1]$ and $h(\cdot)$ is certifiably robust at x with margin $\frac{1-\alpha}{\alpha}$ and radius r, then the mixed classifier $h^\alpha(\cdot)$ is robust in the sense that $\arg\max_i h_i^\alpha(x+\delta) = \arg\max_i h_i(x)$ for all $\delta \in \mathbb{R}^d$ such that $\|\delta\|_p \le r$.*

Proof Suppose that $h(\cdot)$ is certifiably robust at x with margin $\frac{1-\alpha}{\alpha}$ and radius r. Since $\alpha \in [\frac{1}{2}, 1]$, it holds that $\frac{1-\alpha}{\alpha} \in [0,1]$. Let $y = \arg\max_i h_i(x)$. Consider an arbitrary $i \in [c] \setminus \{y\}$ and $\delta \in \mathbb{R}^d$ such that $\|\delta\|_p \le r$. Since $g_i(x+\delta) \in [0,1]$, it holds that

$$\begin{aligned}
&\exp\big(h_y^\alpha(x+\delta)\big) - \exp\big(h_i^\alpha(x+\delta)\big)\\
&\qquad = (1-\alpha)(g_y(x+\delta) - g_i(x+\delta)) + \alpha(h_y(x+\delta) - h_i(x+\delta))\\
&\qquad \ge (1-\alpha)(0-1) + \alpha(h_y(x+\delta) - h_i(x+\delta))\\
&\qquad \ge (\alpha-1) + \alpha\left(\tfrac{1-\alpha}{\alpha}\right) = 0.
\end{aligned}$$

Thus, it holds that $h_y^\alpha(x+\delta) \ge h_i^\alpha(x+\delta)$ for all $i \ne y$, and thus $\arg\max_i h_i^\alpha(x+\delta) = y =$

$\arg\max_i h_i(x)$. ■

Intuitively, Definition 1 ensures that all points within a radius from a nominal point have the same prediction as the nominal point, with the difference between the top and runner-up probabilities no smaller than a threshold. For practical classifiers, the robust margin can be straightforwardly estimated by calculating the confidence gap between the predicted and the runner-up classes at an adversarial input obtained with strong attacks.

While most existing provably robust results consider the special case with zero margin, we will show that models built via common methods are also robust with non-zero margins. We specifically consider two types of popular robust classifiers: Lipschitz continuous models (Theorem 4) and RS models (Theorem 5). Here, Lemma 2 builds the foundation for proving these two theorems, which amounts to showing that Lipschitz and RS models are robust with non-zero margins and thus the mixed classifiers built with them are robust.

Lemma 2 provides further justifications for using probabilities instead of logits in the mixing operation. Intuitively, it holds that $(1-\alpha)g_i(\cdot)$ is bounded between 0 and $1-\alpha$, so as long as α is relatively large (specifically, at least $\frac{1}{2}$), the detrimental effect of $g(\cdot)$'s probabilities when subject to attack can be bounded and be overcome by $h(\cdot)$. Had we used the logits for $g_i(\cdot)$, since this quantity cannot be bounded, it would have been much harder to overcome the vulnerability of $g(\cdot)$.

Since we do not make assumptions on the Lipschitzness or robustness of $g(\cdot)$, Lemma 2 is tight. To understand this, we suppose that there exists some $i \in [c]\backslash\{y\}$ and $\delta \neq 0$ such that $\|\delta\|_p \leq r$ that make $h_y(x+\delta) - h_i(x+\delta) := h_d$ smaller than $\frac{1-\alpha}{\alpha}$, indicating that $-\alpha h_d > \alpha - 1$. Since the only information about $g(\cdot)$ is that $g_i(x+\delta) \in [0,1]$ and thus the value $g_y(x+\delta) - g_i(x+\delta)$ can be any number in $[-1,1]$, it is possible that $(1-\alpha)\left(g_y(x+\delta) - g_i(x+\delta)\right)$ is smaller than $-\alpha h_d$. In this case, it holds that $h^\alpha_y(x+\delta) < h^\alpha_i(x+\delta)$, and thus $\arg\max_i h^\alpha_i(x+\delta) \neq \arg\max_i h_i(x)$.

Definition 3 *A function* $f\colon \mathbb{R}^d \to \mathbb{R}$ *is called* ℓ_p-Lipschitz continuous *if there exists* $L \in (0,\infty)$ *such that* $|f(x') - f(x)| \leq L\|x' - x\|_p$ *for all* $x', x \in \mathbb{R}^d$. *The* ***Lipschitz constant*** *of such* f *is defined to be* $\mathrm{Lip}_p(f) := \inf\{L \in (0,\infty) : |f(x') - f(x)| \leq L\|x'-x\|_p \textit{ for all } x', x \in \mathbb{R}^d\}$.

Assumption 1 *The classifier* $h(\cdot)$ *is robust in the sense that, for all* $i \in \{1, 2, \dots, n\}$, $h_i(\cdot)$ *is* ℓ_p*-Lipschitz continuous with Lipschitz constant* $\mathrm{Lip}_p(h_i)$.

Theorem 4 *Suppose that Assumption 1 holds, and let* $x \in \mathbb{R}^d$ *be arbitrary. Let* $y = \arg\max_i h_i(x)$. *Then, if* $\alpha \in [\frac{1}{2}, 1]$, *it holds that* $\arg\max_i h^\alpha_i(x+\delta) = y$ *for all* $\delta \in \mathbb{R}^d$ *such that*

$$\left\|\delta\right\|_p \leq r^\alpha_p(x) := \min_{i \neq y} \frac{\alpha\left(h_y(x) - h_i(x)\right) + \alpha - 1}{\alpha\left(\mathrm{Lip}_p(h_y) + \mathrm{Lip}_p(h_i)\right)}. \tag{5}$$

Proof Suppose that $\alpha \in [\frac{1}{2}, 1]$, and let $\delta \in \mathbb{R}^d$ be such that $\|\delta\|_p \leq r^\alpha_p(x)$. Furthermore, let $i \in [c] \setminus \{y\}$. It holds that

$$\begin{aligned} h_y(x+\delta) - h_i(x+\delta) &= h_y(x) - h_i(x) + h_y(x+\delta) - h_y(x) + h_i(x) - h_i(x+\delta) \\ &\geq h_y(x) - h_i(x) - \mathrm{Lip}_p(h_y)\|\delta\|_p - \mathrm{Lip}_p(h_i)\|\delta\|_p \\ &\geq h_y(x) - h_i(x) - \left(\mathrm{Lip}_p(h_y) + \mathrm{Lip}_p(h_i)\right) r^\alpha_p(x) \geq \tfrac{1-\alpha}{\alpha}. \end{aligned}$$

Therefore, $h(\cdot)$ is certifiably robust at x with margin $\frac{1-\alpha}{\alpha}$ and radius $r^\alpha_p(x)$. Hence, by Lemma 2, the claim holds. ■

We remark that the ℓ_p norm that Theorem 4 certifies may be arbitrary (e.g., ℓ_1, ℓ_2, or ℓ_∞), so long as the Lipschitz constant of the robust network $h(\cdot)$ is computed with respect to the same norm.

Assumption 1 is not restrictive in practice. For example, Gaussian RS with smoothing variance $\sigma^2 I_d$ yields robust models with ℓ_2-Lipschitz constant $\sqrt{2/\pi\sigma^2}$ (Salman et al., 2019). Moreover, empirically robust methods such as AT and TRADES often train locally Lipschitz continuous models, even though there may not be closed-form theoretical guarantees.

Assumption 1 can be relaxed to the even less restrictive scenario of using local Lipschitz constants over a neighborhood (e.g., a norm ball) around a nominal input x (i.e., how flat $h(\cdot)$ is near x) as a surrogate for the global Lipschitz constants. In this case, Theorem 4 holds for all δ within this neighborhood. Specifically, suppose that for an arbitrary input x and an ℓ_p attack radius ϵ, it holds that $h_y(x) - h_y(x+\delta) \le \epsilon \cdot \mathrm{Lip}_p^x(h_y)$ and $h_i(x+\delta) - h_i(x) \le \epsilon \cdot \mathrm{Lip}_p^x(h_i)$ for all $i \ne y$ and all perturbations δ such that $\|\delta\|_p \le \epsilon$. Furthermore, suppose that the robust radius $r_p^\alpha(x)$, as defined in (5) but use the local Lipschitz constant Lip_p^x as a surrogate to the global constant Lip_p, is not smaller than ϵ. Then, if the robust base classifier $h(\cdot)$ is correct at the nominal point x, then the mixed classifier $h^\alpha(\cdot)$ is robust at x within the radius ϵ. The proof follows that of Theorem 4.

The relaxed Lipschitzness defined above can be estimated for practical differentiable classifiers via an algorithm similar to the PGD attack (Yang et al., 2020). Yang et al. (2020) also showed that many existing empirically robust models, including those trained with AT or TRADES, are in fact locally Lipschitz. Note that Yang et al. (2020) evaluated the local Lipschitz constants of the logits, whereas we analyze the probabilities, whose Lipschitz constants are much smaller. Therefore, Theorem 4 provides important insights into the empirical robustness of the mixed classifier.

An intuitive explanation of Theorem 4 is that if $\alpha \to 1$, then $r_p^\alpha(x) \to \min_{i \ne y} \frac{h_y(x) - h_i(x)}{\mathrm{Lip}_p(h_y) + \mathrm{Lip}_p(h_i)}$, which is the standard Lipschitz-based robust radius of $h(\cdot)$ around x (see (Fazlyab et al., 2019; Hein and Andriushchenko, 2017) for further discussions on Lipschitz-based robustness). On the other hand, if α is too small in comparison to the relative confidence of $h(\cdot)$ and put an excess weight into the non-robust classifier $g(\cdot)$, namely, if there exists $i \ne y$ such that $\alpha \le \frac{1}{1 + h_y(x) - h_i(x)}$, then $r_p^\alpha(x) \le 0$, and in this case, we cannot provide non-trivial certified robustness for $h^\alpha(\cdot)$. If $h(\cdot)$ is 100% confident in its prediction, then $h_y(x) - h_i(x) = 1$ for all $i \ne y$, and therefore this threshold value of α becomes $\frac{1}{2}$, leading to non-trivial certified radii for $\alpha > \frac{1}{2}$. However, once we put over $\frac{1}{2}$ of the weight into $g(\cdot)$, a nonzero radius around x is no longer certifiable. Since no assumptions on the robustness of $g(\cdot)$ around x have been made, this is intuitively the best one can expect.

We now move on to tightening the certified radius in the special case when $h(\cdot)$ is an RS classifier and our robust radii are defined in terms of the ℓ_2 norm.

Assumption 2 *The classifier $h(\cdot)$ is a (Gaussian) randomized smoothing classifier, i.e., $h(x) = \mathbb{E}_{\xi \sim \mathcal{N}(0, \sigma^2 I_d)} \left[\overline{h}(x+\xi) \right]$ for all $x \in \mathbb{R}^d$, where $\overline{h} \colon \mathbb{R}^d \to [0,1]^c$ is a neural model that is non-robust in general. Furthermore, for all $i \in [c]$, $\overline{h}_i(\cdot)$ is not 0 almost everywhere or 1 almost everywhere.*

Theorem 5 *Suppose that Assumption 2 holds, and let $x \in \mathbb{R}^d$ be arbitrary. Let $y = \arg\max_i h_i(x)$ and $y' = \arg\max_{i \ne y} h_i(x)$. Then, if $\alpha \in [\frac{1}{2}, 1]$, it holds that $\arg\max_i h_i^\alpha(x+\delta) = y$ for all $\delta \in \mathbb{R}^d$ such that*

$$\|\delta\|_2 \le r_\sigma^\alpha(x) \coloneqq \frac{\sigma}{2} \Big(\Phi^{-1}\left(\alpha h_y(x)\right) - \Phi^{-1}\left(\alpha h_{y'}(x) + 1 - \alpha\right) \Big).$$

The proof of Theorem 5 is provided in Appendix B in the supplementary materials.

To summarize our certified radii, Theorem 4 applies to very general Lipschitz continuous robust base classifiers $h(\cdot)$ and arbitrary ℓ_p norms, whereas Theorem 5, applying to the ℓ_2 norm and RS base classifiers, strengthens the certified radius by exploiting the stronger Lipschitzness arising from the special structure and smoothness granted by Gaussian convolution operations. Theorems 4 and 5 guarantee that our proposed robustification cannot be easily circumvented by adaptive attacks.

4. Numerical Experiments

4.1. α's Influence on Mixed Classifier Robustness

We first use the CIFAR-10 dataset to evaluate the mixed classifier $h^{\alpha}(\cdot)$ with various values of α. We use a ResNet18 model trained on unattacked images as the standard base model $g(\cdot)$ and use another ResNet18 trained on PGD$_{20}$ data as the robust base model $h(\cdot)$. We consider PGD$_{20}$ attacks that target $g(\cdot)$ and $h(\cdot)$ individually (abbreviated as STD and ROB attacks and can be regarded as transfer attacks), in addition to the adaptive PGD$_{20}$ attack generated using the end-to-end gradient of $h^{\alpha}(\cdot)$, denoted as the MIX attack.

The test accuracy of each mixed classifier is presented in Figure 2. As α increases, the clean accuracy of $h^{\alpha}(\cdot)$ converges from the clean accuracy of $g(\cdot)$ to the clean accuracy of $h(\cdot)$. In terms of attacked performance, when the attack targets $g(\cdot)$, the attacked accuracy increases with α. When the attack targets $h(\cdot)$, the attacked accuracy decreases with α, showing that the attack targeting $h(\cdot)$ becomes more benign when the mixed classifier emphasizes $g(\cdot)$. When the attack targets the mixed classifier $h^{\alpha}(\cdot)$, the attacked accuracy increases with α.

When α is around 0.5, the MIX-attacked accuracy of $h^{\alpha}(\cdot)$ quickly increases from near zero to more than 30% (two-thirds of $h(\cdot)$'s attacked accuracy). This observation precisely matches the theoretical intuition from Theorem 4. Meanwhile, when α is greater than 0.5, the clean accuracy gradually decreases at a much slower rate, leading to the alleviated accuracy-robustness trade-off.

4.2. The Relationship between $h^{\alpha}(\cdot)$'s Robustness and $h(\cdot)$'s Confidence

This difference in how clean and attacked accuracy change with α can be explained by the prediction confidence of the robust base classifier $h(\cdot)$. Specifically, Table 1 confirms that $h(\cdot)$ makes confident correct predictions even when under attack (average robust margin is 0.768). Moreover, $h(\cdot)$'s robust margin follows a long-tail distribution: the median robust margin is 0.933, much larger than the 0.768 mean. Thus, most attacked inputs correctly classified by $h(\cdot)$ are highly confident (i.e., robust with large margins). As Lemma 2 suggests, such a property is precisely what the mixed classifier relies on. Intuitively, once α becomes greater than 0.5 and gives $h(\cdot)$ more authority over $g(\cdot)$, $h(\cdot)$ can use its confidence to correct $g(\cdot)$'s mistakes under attack.

On the other hand, $h(\cdot)$ is unconfident when producing incorrect predictions on clean data, with the top two classes' output probabilities separated by merely 0.434. This probability gap again forms a long-tail distribution (the median is 0.378 which is less than the mean), confirming that $h(\cdot)$ rarely makes confident incorrect predictions. Now, consider clean data that $g(\cdot)$ correctly classifies and $h(\cdot)$ mispredicts. Recall that we assume $g(\cdot)$ to be more accurate but less robust, so this scenario should be common. Since $g(\cdot)$ is confident (average top two classes probability gap is 0.982) and $h(\cdot)$ is usually unconfident, even when $\alpha > 0.5$ and $g(\cdot)$ has less authority than $h(\cdot)$ in the mixture, $g(\cdot)$ can still correct some of the mistakes from $h(\cdot)$.

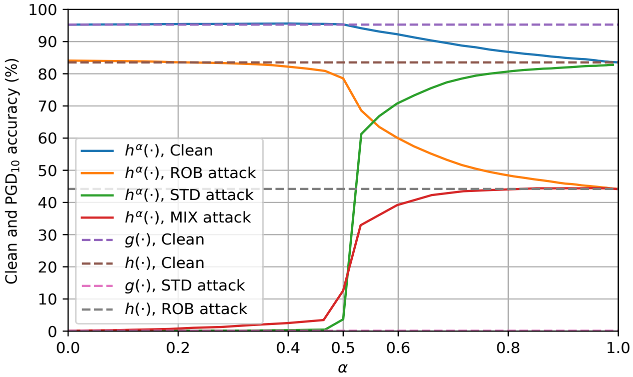


Figure 2: The accuracy of the mixed classifier $h^{\alpha}(\cdot)$ at various α values. "STD attack", "ROB attack", and "MIX attack" refer to the PGD$_{20}$ attack generated using the gradient of $g(\cdot)$, $h(\cdot)$, and $h^{\alpha}(\cdot)$ respectively, with ϵ set to $\frac{8}{255}$.

Table 1: Average gap between the probabilities of the predicted class and the runner-up class.

	Clean Instances	
	Correct	Incorrect
$g(\cdot)$	0.982	0.698
$h(\cdot)$	0.854	0.434
	PGD$_{20}$ Instances	
	Correct	Incorrect
$g(\cdot)$	0.602	0.998
$h(\cdot)$	0.768	0.635

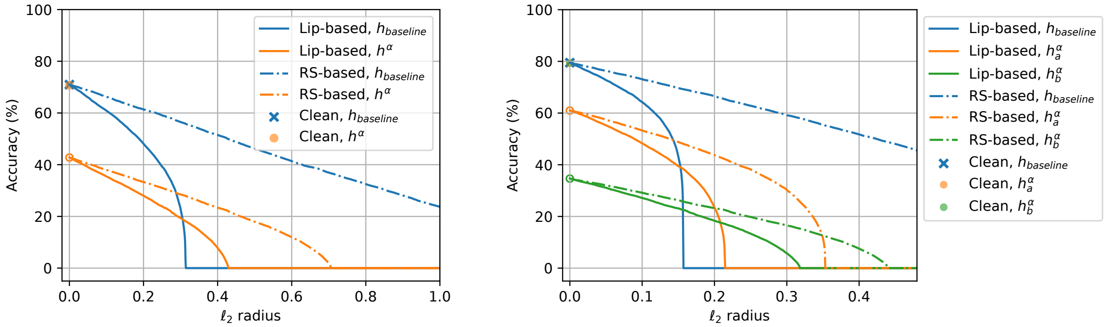
In summary, $h(\cdot)$ is confident when making correct predictions on attacked data while being unconfident when misclassifying clean data, and such a confidence property is the key source of the mixed classifier's improved accuracy-robustness trade-off. Additional analyses in Appendix A with alternative base models imply that multiple existing robust classifiers share this benign confidence property and thus help the mixed classifier improve the trade-off.

4.3. Visualization of the Certified Robust Radii

Next, we visualize the certified robust radii presented in Theorem 4 and Theorem 5. Since a (Gaussian) RS model with smoothing covariance matrix $\sigma^2 I_d$ has an ℓ_2-Lipschitz constant $\sqrt{2/\pi\sigma^2}$, such a model can be used to simultaneously visualize both theorems, with Theorem 5 giving tighter certificates of robustness. Note that RS models with a larger smoothing variance certify larger radii but achieve lower clean accuracy, and vice versa. Here, we consider the CIFAR-10 dataset and select $g(\cdot)$ to be a ConvNeXT-T model with a clean accuracy of 97.25%, and use the RS models presented in (Zhang et al., 2019) as $h(\cdot)$. For a fair comparison, we select an α value such that the clean accuracy of the constructed mixed classifier $h^{\alpha}(\cdot)$ matches that of another RS model $h_{\text{baseline}}(\cdot)$ with a smaller smoothing variance. The expectation term in the RS formulation is approximated with the empirical mean of 10000 random perturbations drawn from $\mathcal{N}(0, \sigma^2 I_d)$, and the certified radii of $h_{\text{baseline}}(\cdot)$ are calculated using Theorems 4 and 5 by setting α to 1. Figure 3 displays the calculated certified accuracy of $h^{\alpha}(\cdot)$ and $h_{\text{baseline}}(\cdot)$ at various attack radii. The ordinate "Accuracy" at a given abscissa "ℓ_2 radius" reflects the percentage of the test data for which the considered model gives a correct prediction as well as a certified radius at least as large as the ℓ_2 radius under consideration.

In both subplots of Figure 3, the certified robustness curves of $h^{\alpha}(\cdot)$ do not connect to the clean accuracy when $\alpha \to 0$. This is because Theorems 4 and 5 both consider robustness with respect to $h(\cdot)$ and do not certify test inputs at which $h(\cdot)$ makes incorrect predictions, even though $h^{\alpha}(\cdot)$ may correctly predict some of these points. This is reasonable because we do not assume any robustness or Lipschitzness of $g(\cdot)$, and $g(\cdot)$ is allowed to be arbitrarily incorrect whenever the radius is non-zero.

The Lipschitz-based bound of Theorem 4 allows us to visualize the performance of the mixed classifier $h^{\alpha}(\cdot)$ when $h(\cdot)$ is an ℓ_2-Lipschitz model. In this case, the curves associated with $h^{\alpha}(\cdot)$



(a) $h_{\text{baseline}}(\cdot)$: RS with $\sigma = 0.5$.
$h^\alpha(\cdot)$: $\alpha = 0.76$; $h(\cdot)$ is RS with $\sigma = 1$.

(b) $h_{\text{baseline}}(\cdot)$: RS with $\sigma = 0.25$.
Consider two mixed classifier examples:
$h_a^\alpha(\cdot)$: $\alpha = 0.76$; $h_a(\cdot)$ is RS with $\sigma = 0.5$;
$h_b^\alpha(\cdot)$: $\alpha = 0.67$; $h_b(\cdot)$ is RS with $\sigma = 1.0$.

Figure 3: Comparing the certified accuracy-robustness trade-off of RS models and our mixed classifier using both Lipschitz-based (Lip-based) certificates and RS-based certificates (Theorems 4 and 5, respectively). The clean accuracy is the same between $h_{\text{baseline}}(\cdot)$ and $h^\alpha(\cdot)$ in each subfigure, and the empty circles represent discontinuity in the certified accuracy at radius 0.

and $h_{\text{baseline}}(\cdot)$ intersect, with $h^\alpha(\cdot)$ achieving higher certified accuracy at larger radii and $h_{\text{baseline}}(\cdot)$ certifying more points at smaller radii. Adjusting α and the Lipschitz constant of $h(\cdot)$ can change the location of this intersection while maintaining the clean accuracy. Thus, the mixed classifier allows for optimizing the certified accuracy at a particular radius without sacrificing clean accuracy.

The RS-based bound from Theorem 5 captures the behavior of the mixed classifier $h^\alpha(\cdot)$ when $h(\cdot)$ is an RS model. For both $h^\alpha(\cdot)$ and $h_{\text{baseline}}(\cdot)$, the RS-based bounds certify larger radii than the corresponding Lipschitz-based bounds. Nonetheless, $h_{\text{baseline}}(\cdot)$ can certify more points with the RS-based guarantee. Intuitively, this phenomenon suggests that RS models can yield correct but low-confidence predictions when under large-radius attack, and thus may not be best-suited for our mixing operation, which relies on robustness with non-zero margins. Meanwhile, Lipschitz models, a more general and common class of models, exploit the mixing operation more effectively. Moreover, as shown in Figure 2 and Table 1, empirically robust models often yield high-confidence correct predictions when under attack, making them more suitable to be used as $h^\alpha(\cdot)$'s robust base classifier.

5. Conclusions

This work proposes to mix the predicted probabilities of an accurate classifier and a robust classifier to mitigate the accuracy-robustness trade-off. These two base classifiers can be pre-trained, and the resulting mixed classifier requires no additional training. Theoretical results certify that the mixed classifier inherits the robustness of the robust base model under realistic assumptions. Empirical evaluations show that our method approaches the high accuracy of the latest standard models while retaining the robustness of modern robust classification methods. Hence, this work provides a foundation for future research to focus on either accuracy or robustness without sacrificing the other, providing additional incentives for deploying robust models in safety-critical control.

Acknowledgments

This work was supported by grants from ONR, NSF, and C3 AI.

References

Manaar Alam, Shubhajit Datta, Debdeep Mukhopadhyay, Arijit Mondal, and Partha Pratim Chakrabarti. Resisting adversarial attacks in deep neural networks using diverse decision boundaries. *arXiv preprint arXiv:2208.08697*, 2022.

Brendon Anderson, Ziye Ma, Jingqi Li, and Somayeh Sojoudi. Tightened convex relaxations for neural network robustness certification. In *IEEE Conference on Decision and Control*, 2020.

Brendon G Anderson and Somayeh Sojoudi. Data-driven certification of neural networks with random input noise. *IEEE Transactions on Control of Network Systems*, 2022a.

Brendon G. Anderson and Somayeh Sojoudi. Certified robustness via locally biased randomized smoothing. In *Learning for Dynamics and Control Conference*, 2022b.

Yatong Bai, Tanmay Gautam, Yu Gai, and Somayeh Sojoudi. Practical convex formulation of robust one-hidden-layer neural network training. *American Control Conference*, 2022a.

Yatong Bai, Tanmay Gautam, and Somayeh Sojoudi. Efficient global optimization of two-layer ReLU networks: Quadratic-time algorithms and adversarial training. *SIAM Journal on Mathematics of Data Science*, 2022b.

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

Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D. Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, et al. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*, 2016.

Tianlong Chen, Sijia Liu, Shiyu Chang, Yu Cheng, Lisa Amini, and Zhangyang Wang. Adversarial robustness: From self-supervised pre-training to fine-tuning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020.

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

Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. Robust physical-world attacks on deep learning visual classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

Lijie Fan, Sijia Liu, Pin-Yu Chen, Gaoyuan Zhang, and Chuang Gan. When does contrastive learning preserve adversarial robustness from pretraining to finetuning? In *Advances in Neural Information Processing Systems*, 2021.

Mahyar Fazlyab, Alexander Robey, Hamed Hassani, Manfred Morari, and George Pappas. Efficient and accurate estimation of Lipschitz constants for deep neural networks. In *Advances in Neural Information Processing Systems*, 2019.

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

Sven Gowal, Sylvestre-Alvise Rebuffi, Olivia Wiles, Florian Stimberg, Dan A. Calian, and Timothy Mann. Improving robustness using generated data. *arXiv preprint arXiv:2110.09468*, 2021.

Matthias Hein and Maksym Andriushchenko. Formal guarantees on the robustness of a classifier against adversarial manipulation. In *Advances in Neural Information Processing Systems*, 2017.

Ting-Kuei Hu, Tianlong Chen, Haotao Wang, and Zhangyang Wang. Triple wins: Boosting accuracy, robustness and efficiency together by enabling input-adaptive inference. In *International Conference on Learning Representations*, 2020.

Sandy H. Huang, Nicolas Papernot, Ian J. Goodfellow, Yan Duan, and Pieter Abbeel. Adversarial attacks on neural network policies. In *International Conference on Learning Representations*, 2017.

Xiaojun Jia, Yong Zhang, Baoyuan Wu, Ke Ma, Jue Wang, and Xiaochun Cao. LAS-AT: Adversarial training with learnable attack strategy. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2022.

Alex Krizhevsky. Learning multiple layers of features from tiny images, 2012. URL https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.

Aounon Kumar, Alexander Levine, and Soheil Feizi. Policy smoothing for provably robust reinforcement learning. In *International Conference on Learning Representations*, 2022.

Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. Adversarial machine learning at scale. In *International Conference on Learning Representations*, 2017.

Alex Lamb, Vikas Verma, Juho Kannala, and Yoshua Bengio. Interpolated adversarial training: Achieving robust neural networks without sacrificing too much accuracy. In *ACM Workshop on Artificial Intelligence and Security*, 2019.

Alexander Levine, Sahil Singla, and Soheil Feizi. Certifiably robust interpretation in deep learning. *arXiv preprint arXiv:1905.12105*, 2019.

Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. End-to-end training of deep visuomotor policies. *The Journal of Machine Learning Research*, 17(1):1334–1373, 2016.

Bai Li, Changyou Chen, Wenlin Wang, and Lawrence Carin. Certified adversarial robustness with additive noise. In *Advances in Neural Information Processing Systems*, 2019.

Aishan Liu, Xianglong Liu, Jiaxin Fan, Yuqing Ma, Anlan Zhang, Huiyuan Xie, and Dacheng Tao. Perceptual-sensitive GAN for generating adversarial patches. In *The AAAI Conference on Artificial Intelligence*, 2019.

Xuanqing Liu, Minhao Cheng, Huan Zhang, and Cho-Jui Hsieh. Towards robust neural networks via random self-ensemble. In *European Conference on Computer Vision*, 2018.

Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A ConvNet for the 2020s. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2022.

Ziye Ma and Somayeh Sojoudi. A sequential framework towards an exact SDP verification of neural networks. In *International Conference on Data Science and Advanced Analytics*, 2021.

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

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

Tianyu Pang, Kun Xu, Chao Du, Ning Chen, and Jun Zhu. Improving adversarial robustness via promoting ensemble diversity. In *International Conference on Machine Learning*, 2019.

Tianyu Pang, Min Lin, Xiao Yang, Jun Zhu, and Shuicheng Yan. Robustness and accuracy could be reconcilable by (proper) definition. *arXiv preprint arXiv:2202.10103*, 2022.

Samuel Pfrommer, Brendon G Anderson, and Somayeh Sojoudi. Projected randomized smoothing for certified adversarial robustness. *Transactions on Machine Learning Research*, 2023.

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

Sylvestre-Alvise Rebuffi, Sven Gowal, Dan A Calian, Florian Stimberg, Olivia Wiles, and Timothy Mann. Fixing data augmentation to improve adversarial robustness. *arXiv preprint arXiv:2103.01946*, 2021.

Hadi Salman, Jerry Li, Ilya Razenshteyn, Pengchuan Zhang, Huan Zhang, Sebastien Bubeck, and Greg Yang. Provably robust deep learning via adversarially trained smoothed classifiers. *Advances in Neural Information Processing Systems*, 2019.

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

Vikash Sehwag, Saeed Mahloujifar, Tinashe Handina, Sihui Dai, Chong Xiang, Mung Chiang, and Prateek Mittal. Robust learning meets generative models: Can proxy distributions improve adversarial robustness? In *International Conference on Learning Representations*, 2022.

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

Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction.* MIT press, 2018.

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

Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian J. Goodfellow, Dan Boneh, and Patrick D. McDaniel. Ensemble adversarial training: Attacks and defenses. In *International Conference on Learning Representations*, 2018.

Florian Tramèr, Nicholas Carlini, Wieland Brendel, and Aleksander Madry. On adaptive attacks to adversarial example defenses. In *Advances in Neural Information Processing Systems*, 2020.

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

Bichen Wu, Forrest Iandola, Peter H. Jin, and Kurt Keutzer. SqueezeDet: Unified, small, low power fully convolutional neural networks for real-time object detection for autonomous driving. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017.

Fan Wu, Linyi Li, Zijian Huang, Yevgeniy Vorobeychik, Ding Zhao, and Bo Li. CROP: Certifying robust policies for reinforcement learning through functional smoothing. In *International Conference on Learning Representations*, 2022.

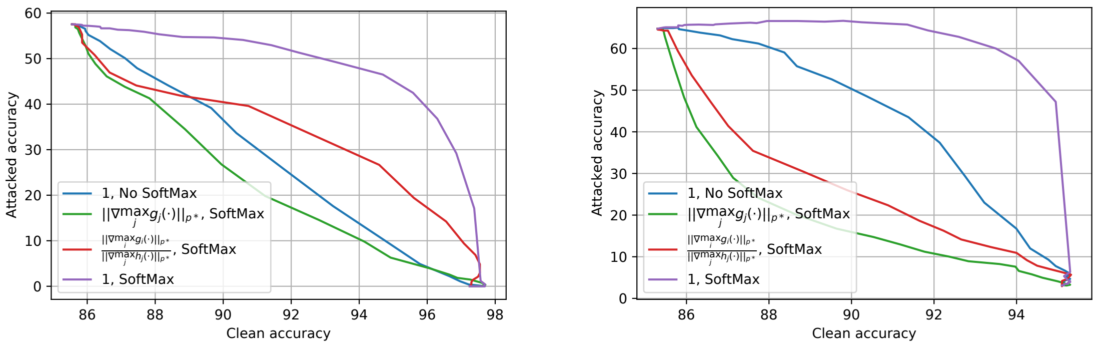
Yao-Yuan Yang, Cyrus Rashtchian, Hongyang Zhang, Russ R. Salakhutdinov, and Kamalika Chaudhuri. A closer look at accuracy vs. robustness. In *Annual Conference on Neural Information Processing Systems*, 2020.

Haichao Zhang and Jianyu Wang. Defense against adversarial attacks using feature scattering-based adversarial training. In *Annual Conference on Neural Information Processing Systems*, 2019.

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

Haizhong Zheng, Ziqi Zhang, Juncheng Gu, Honglak Lee, and Atul Prakash. Efficient adversarial training with transferable adversarial examples. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020.

Yaowei Zheng, Richong Zhang, and Yongyi Mao. Regularizing neural networks via adversarial model perturbation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2021.



(a) ConvNeXT-T and TRADES WRN-34 under ℓ_∞ PGD attack. (b) Standard and AT ResNet18s under ℓ_2 PGD attack.

Figure 4: Comparing the options for $R_i(x)$ with alternative selections of base classifiers.

Table 2: Experiment settings for comparing the choices of $R_i(x)$.

	Attack Budget; PGD Steps	$g(\cdot)$ Architecture	$h(\cdot)$ Architecture
Figure 1	$\ell_\infty, \ \epsilon = \frac{8}{255}$, 10 Steps	Standard ResNet18	ℓ_∞-adversarially-trained ResNet18
Figure 4a	$\ell_\infty, \ \epsilon = \frac{8}{255}$, 20 Steps	Standard ConvNeXT-T	TRADES WideResNet-34
Figure 4b	$\ell_2, \ \ \epsilon = 0.5$, 20 Steps	Standard ResNet18	ℓ_2-adversarially-trained ResNet18

Appendix A. Additional Empirical Support for $R_i(x) = 1$

Finally, we use additional empirical evidence (Figures 4a and 4b) to show that $R_i(x) = 1$ is the appropriate choice for the mixed classifier and that the probabilities should be used for the mixture. While most experiments in this paper are based on the popular ResNet architecture, our method does not depend on any ResNet properties. Therefore, for the experiment in Figure 4a, we select a more modern ConvNeXT-T model (Liu et al., 2022) pre-trained on ImageNet-1k as an alternative architecture for $g(\cdot)$. We also use a robust model trained via TRADES in place of an adversarially-trained network for $h(\cdot)$ for the interest of diversity. Additionally, although most of our experiments are based on ℓ_∞ attacks, the proposed method applies to all ℓ_p attack budgets. In Figure 4b, we provide an example that considers the ℓ_2 attack. The experiment settings are summarized in Table 2.

Figures 4a and 4b confirm that setting $R_i(x)$ to the constant 1 achieves the best trade-off curve between clean and attacked accuracy, and that mixing the probabilities outperforms mixing the logits. This result aligns with the conclusions of Figure 1 and our theoretical analyses.

For all three cases listed in Table 2, the mixed classifier reduces the error rate of $h(\cdot)$ on clean data by half while maintaining 80% of $h(\cdot)$'s attacked accuracy. This observation suggests that the mixed classifier noticeably alleviates the accuracy-robustness trade-off. Additionally, our method is especially suitable for applications where the clean accuracy gap between $g(\cdot)$ and $h(\cdot)$ is large. On easier datasets such as MNIST and CIFAR-10, this gap has been greatly reduced by the latest advancements in constructing robust classifiers. However, on harder tasks such as CIFAR-100 and ImageNet-1k, this gap is still large, even for state-of-the-art methods. For these applications, standard classifiers often benefit much more from pre-training on larger datasets than robust models.

Appendix B. Proof of Theorem 5

Theorem 5 (Restated) *Suppose that Assumption 2 holds, and let $x \in \mathbb{R}^d$ be arbitrary. Let $y = \arg\max_i h_i(x)$ and $y' = \arg\max_{i \neq y} h_i(x)$. Then, if $\alpha \in [\frac{1}{2}, 1]$, it holds that $\arg\max_i h_i^\alpha(x+\delta) = y$ for all $\delta \in \mathbb{R}^d$ such that*

$$\|\delta\|_2 \leq r_\sigma^\alpha(x) \coloneqq \frac{\sigma}{2}\Big(\Phi^{-1}\left(\alpha h_y(x)\right) - \Phi^{-1}\left(\alpha h_{y'}(x) + 1 - \alpha\right)\Big).$$

Proof First, note that since every $\overline{h}_i(\cdot)$ is not 0 almost everywhere or 1 almost everywhere, it holds that $h_i(x) \in (0,1)$ for all i and all x. Now, suppose that $\alpha \in [\frac{1}{2}, 1]$, and let $\delta \in \mathbb{R}^d$ be such that $\|\delta\|_2 \leq r_\sigma^\alpha(x)$. Let $\mu_\alpha \coloneqq \frac{1-\alpha}{\alpha}$. Define the function $\tilde{h} \colon \mathbb{R}^d \to \mathbb{R}^c$ by

$$\tilde{h}_y(x) = \frac{\overline{h}_y(x)}{1+\mu_\alpha}, \quad \tilde{h}_i(x) = \frac{\overline{h}_i(x) + \mu_\alpha}{1+\mu_\alpha} \text{ for all } i \neq y.$$

Furthermore, define $\hat{h} \colon \mathbb{R}^d \to \mathbb{R}^c$ by $\hat{h}(x) = \mathbb{E}_{\xi \sim \mathcal{N}(0,\sigma^2 I_d)}\left[\tilde{h}(x+\xi)\right]$.

Then, since $\tilde{h}_y(x) = \frac{\overline{h}_y(x)}{1+\mu_\alpha} \in (0, \frac{1}{1+\mu_\alpha}) \subseteq (0,1)$ and $\tilde{h}_i(x) = \frac{\overline{h}_i(x)+\mu_\alpha}{1+\mu_\alpha} \in (\frac{\mu_\alpha}{1+\mu_\alpha}, 1) \subseteq (0,1)$ for all $i \neq y$, it must be the case that $0 < \hat{h}_i(x) < 1$ for all i and all x, and hence, for all i, the function $x \mapsto \Phi^{-1}\big(\hat{h}_i(x)\big)$ is ℓ_2-Lipschitz continuous with Lipschitz constant $\frac{1}{\sigma}$ (see (Levine et al., 2019, Lemma 1), or Lemma 2 in (Salman et al., 2019) and the discussion thereafter). Therefore,

$$\left|\Phi^{-1}\big(\hat{h}_i(x+\delta)\big) - \Phi^{-1}\big(\hat{h}_i(x)\big)\right| \leq \frac{\|\delta\|_2}{\sigma} \leq \frac{r_\sigma^\alpha(x)}{\sigma} \tag{6}$$

for all i. Applying (6) for $i = y$ yields that

$$\Phi^{-1}\big(\hat{h}_y(x+\delta)\big) \geq \Phi^{-1}\big(\hat{h}_y(x)\big) - \frac{r_\sigma^\alpha(x)}{\sigma}. \tag{7}$$

Since Φ^{-1} monotonically increases and $\hat{h}_i(x) \leq \hat{h}_{y'}(x)$ for all $i \neq y$, applying (6) to $i \neq y$ gives

$$\Phi^{-1}\big(\hat{h}_i(x+\delta)\big) \leq \Phi^{-1}\big(\hat{h}_i(x)\big) + \frac{r_\sigma^\alpha(x)}{\sigma} \leq \Phi^{-1}\big(\hat{h}_{y'}(x)\big) + \frac{r_\sigma^\alpha(x)}{\sigma}. \tag{8}$$

Subtracting (8) from (7) gives that

$$\Phi^{-1}\big(\hat{h}_y(x+\delta)\big) - \Phi^{-1}\big(\hat{h}_i(x+\delta)\big) \geq \Phi^{-1}\big(\hat{h}_y(x)\big) - \Phi^{-1}\big(\hat{h}_{y'}(x)\big) - \frac{2r_\sigma^\alpha(x)}{\sigma}$$

for all $i \neq y$. By the definitions of μ_α, $r_\sigma^\alpha(x)$, and $\hat{h}(x)$, the right-hand side of this inequality equals zero. Since Φ monotonically increases, we find that $\hat{h}_y(x+\delta) \geq \hat{h}_i(x+\delta)$ for all $i \neq y$. Thus,

$$\begin{aligned} \frac{h_y(x+\delta)}{1+\mu_\alpha} &= \mathbb{E}_{\xi \sim \mathcal{N}(0,\sigma^2 I_d)}\left[\frac{\overline{h}_y(x+\delta+\xi)}{1+\mu_\alpha}\right] = \hat{h}_y(x+\delta) \\ &\geq \hat{h}_i(x+\delta) = \mathbb{E}_{\xi \sim \mathcal{N}(0,\sigma^2 I_d)}\left[\frac{\overline{h}_i(x+\delta+\xi)+\mu_\alpha}{1+\mu_\alpha}\right] = \frac{h_i(x+\delta)+\mu_\alpha}{1+\mu_\alpha}. \end{aligned}$$

Hence, $h_y(x+\delta) \geq h_i(x+\delta) + \mu_\alpha$ for all $i \neq y$, so $h(\cdot)$ is certifiably robust at x with margin $\mu_\alpha = \frac{1-\alpha}{\alpha}$ and radius $r_\sigma^\alpha(x)$. Therefore, by Lemma 2, it holds that $\arg\max_i h_i^\alpha(x+\delta) = y$ for all $\delta \in \mathbb{R}^d$ such that $\|\delta\|_2 \leq r_\sigma^\alpha(x)$, which concludes the proof. ■