Certified Adversarial Robustness via Randomized Smoothing

Jeremy Cohen 1  Elan Rosenfeld 1  J. Zico Kolter 1 2

Abstract
We show how to turn any classifier that classifies well under Gaussian noise into a new classifier that is certifiably robust to adversarial perturbations under the $\ell_2$ norm. While this “randomized smoothing” technique has been proposed before in the literature, we are the first to provide a tight analysis, which establishes a close connection between $\ell_2$ robustness and Gaussian noise. We use the technique to train an ImageNet classifier with e.g. a certified top-1 accuracy of 49% under adversarial perturbations with $\ell_2$ norm less than 0.5 (=127/255). Smoothing is the only approach to certifiably robust classification which has been shown feasible on full-resolution ImageNet. On smaller-scale datasets where competing approaches to certified $\ell_2$ robustness are viable, smoothing delivers higher certified accuracies. The empirical success of the approach suggests that provable methods based on randomization at prediction time are a promising direction for future research into adversarially robust classification. Code and models are available at http://github.com/locuslab/smoothing.

1. Introduction
Modern image classifiers achieve high accuracy on i.i.d. test sets but are not robust to small, adversarially-chosen perturbations of their inputs (Szegedy et al., 2014; Biggio et al., 2013). Given an image $x$ correctly classified by, say, a neural network, an adversary can usually engineer an adversarial perturbation $\delta$ so small that $x + \delta$ looks just like $x$ to the human eye, yet the network classifies $x + \delta$ as a different, incorrect class. Many works have proposed heuristic methods for training classifiers intended to be robust to adversarial perturbations. However, most of these heuristics have been subsequently shown to fail against suitably powerful adversaries (Carlini & Wagner, 2017; Athalye et al., 2018; Uesato et al., 2018). In response, a line of work on certifiable robustness studies classifiers whose prediction at any point $x$ is verifiably constant within some set around $x$ (e.g. Wong & Kolter, 2018; Raghunathan et al., 2018a). In most of these works, the robust classifier takes the form of a neural network. Unfortunately, all existing approaches for certifying the robustness of neural networks have trouble scaling to networks that are large and expressive enough to solve problems like ImageNet.

One workaround is to look for robust classifiers that are not neural networks. In this paper, we analyze an operation we call randomized smoothing which transforms any arbitrary base classifier $f$ into a new “smoothed classifier” $g$ that is certifiably robust in $\ell_2$ norm. Let $f$ be an arbitrary classifier which maps inputs $\mathbb{R}^d$ to classes $\mathcal{Y}$. For any input $x$, the smoothed classifier’s prediction $g(x)$ is defined to be the class which $f$ is most likely to classify the random Gaussian corruption $x + \delta$ of $x$ as. That is, $g(x)$ returns the most probable prediction by $f$ of random Gaussian corruptions of $x$.

If the base classifier $f$ is most likely to classify $N(x, \sigma^2 I)$ as $x$’s correct class, then the smoothed classifier $g$ will be

---

1Carnegie Mellon University 2Bosch Center for AI. Correspondence to: Jeremy Cohen <jeremycohen@cmu.edu>.


---

1We adopt this term because it has been used to describe a similar technique in a different context (Duchi et al., 2012).
correct at $x$. But the smoothed classifier $g$ will also possess a desirable property that the base classifier may lack: one can verify that $g$’s prediction is constant within an $\ell_2$ ball around any input $x$, simply by estimating the probabilities with which $f$ classifies $\mathcal{N}(x, \sigma^2 I)$ as each class. The higher the probability with which $f$ classifies $\mathcal{N}(x, \sigma^2 I)$ as the most probable class, the larger the $\ell_2$ radius around $x$ in which $g$ provably returns that class.

Lecuyer et al. (2019) proposed randomized smoothing as a provable adversarial defense, and used it to train the first certifiably robust classifier for ImageNet. Subsequently, Li et al. (2018) proved a stronger robustness guarantee. However, both of these guarantees are loose, in the sense that the smoothed classifier $g$ is provably always more robust than the guarantee indicates. In this paper, we prove the first tight robustness guarantee for randomized smoothing. Our analysis reveals that smoothing with Gaussian noise naturally induces certifiable robustness under the $\ell_2$ norm. We suspect that other, as-yet-unknown noise distributions might induce robustness to other perturbation sets such as general $\ell_p$ norm balls.

Randomized smoothing has one major drawback. If $f$ is a neural network, it is not possible to exactly compute the probabilities with which $f$ classifies $\mathcal{N}(x, \sigma^2 I)$ as each class. Therefore, it is not possible to exactly evaluate the smoothed classifier $g$ or to exactly compute the radius in which $g$ is robust. Instead, we present Monte Carlo algorithms for both tasks that are guaranteed to succeed with arbitrarily high probability.

Despite this drawback, randomized smoothing enjoys several compelling advantages over other certifiably robust classifiers proposed in the literature: it makes no assumptions about the base classifier’s architecture, it is simple to implement and understand, and, most importantly, it permits the use of arbitrarily large neural networks as the base classifier. In contrast, other certified defenses do not currently scale to large networks. Indeed, smoothing is the only certified adversarial defense which has been shown feasible on the full-resolution ImageNet classification task.

We use randomized smoothing to train state-of-the-art certifiably $\ell_2$-robust ImageNet classifiers; for example, one of them achieves 49% provable top-1 accuracy under adversarial perturbations with $\ell_2$ norm less than 127/255 (Table 1). We also demonstrate that on smaller-scale datasets like CIFAR-10 and SHVN, where competing approaches to certified $\ell_2$ robustness are feasible, randomized smoothing can deliver better certified accuracies, both because it enables the use of larger networks and because it does not constrain the expressivity of the base classifier.

### 2. Related Work

Many works have proposed classifiers intended to be robust to adversarial perturbations. These approaches can be broadly divided into empirical defenses, which empirically seem robust to known adversarial attacks, and certified defenses, which are provably robust to certain kinds of adversarial perturbations.

**Empirical defenses** The most successful empirical defense to date is adversarial training (Goodfellow et al., 2015; Kurakin et al., 2017; Madry et al., 2018), in which adversarial examples are found during training (often using projected gradient descent) and added to the training set. Unfortunately, it is typically impossible to tell whether a prediction by an empirically robust classifier is truly robust to adversarial perturbations; the most that can be said is that a specific attack was unable to find any. In fact, many heuristic defenses proposed in the literature were later “broken” by stronger adversaries (Carlini & Wagner, 2017; Athalye et al., 2018; Uesato et al., 2018; Athalye & Carlini, 2018).

![Figure 2. The smoothed classifier’s prediction at an input $x$ (left) is defined as the most likely prediction by the base classifier on random Gaussian corruptions of $x$ (right; $\sigma = 0.5$). Note that this Gaussian noise is much larger in magnitude than the adversarial perturbations to which $g$ is provably robust. One interpretation of randomized smoothing in high dimension is that these large random perturbations “drown out” small adversarial perturbations.](image)

### Table 1. Approximate certified accuracy on ImageNet. Each row shows a radius $r$, the best hyperparameter $\sigma$ for that radius, the approximate certified accuracy at radius $r$ of the corresponding smoothed classifier, and the standard accuracy of the corresponding smoothed classifier. To give a sense of scale, a perturbation with $\ell_2$ radius 1.0 could change one pixel by 255, ten pixels by 80, 100 pixels by 25, or 1000 pixels by 8. Random guessing on ImageNet would attain 0.1% accuracy.

<table>
<thead>
<tr>
<th>$\ell_2$ radius</th>
<th>best $\sigma$</th>
<th>Cert. Acc (%)</th>
<th>Std. Acc(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>0.25</td>
<td>49</td>
<td>67</td>
</tr>
<tr>
<td>1.0</td>
<td>0.50</td>
<td>37</td>
<td>57</td>
</tr>
<tr>
<td>2.0</td>
<td>0.50</td>
<td>19</td>
<td>57</td>
</tr>
<tr>
<td>3.0</td>
<td>1.00</td>
<td>12</td>
<td>44</td>
</tr>
</tbody>
</table>
Certified Adversarial Robustness via Randomized Smoothing

Aiming to escape this cat-and-mouse game, a growing body of work has focused on defenses with formal guarantees.

**Certified defenses** A classifier is said to be **certifiably robust** if for any input $x$, one can easily obtain a guarantee that the classifier’s prediction is constant within some set around $x$, often an $\ell_2$ or $\ell_\infty$ ball. In most work in this area, the certifiably robust classifier is a neural network. Some works propose algorithms for certifying the robustness of generically trained networks, while others (Wong & Kolter, 2018; Raghunathan et al., 2018a) propose both a robust training method and a complementary certification mechanism.

Certification methods are either **exact** (a.k.a “complete”) or **conservative** (a.k.a “sound but incomplete”). In the context of $\ell_p$ norm-bounded perturbations, exact methods take a classifier $g$, input $x$, and radius $r$, and report whether or not there exists a perturbation $\delta$ within $\|\delta\| \leq r$ for which $g(x) \neq g(x + \delta)$. In contrast, conservative methods either certify that no such perturbation exists or decline to make a certification; they may decline even when it is true that no such perturbation exists. Exact methods are usually based on Satisfiability Modulo Theories (Katz et al., 2017; Carlini et al., 2017; Ehlers, 2017; Huang et al., 2017) or mixed integer linear programming (Cheng et al., 2017; Lomuscio & Maganti, 2017; Dutta et al., 2017; Fischetti & Jo, 2018; Bunel et al., 2018). Unfortunately, no exact methods have been shown to scale beyond moderate-sized (100,000 activations) networks (Tjeng et al., 2019), and networks of that size can only be verified when they are trained in a manner that impairs their expressivity.

Conservative certification is more scalable. Some conservative methods bound the **global** Lipschitz constant of the neural network (Gouk et al., 2018; Tsuzuku et al., 2018; Anil et al., 2019; Cisse et al., 2017), but these approaches tend to be very loose on expressive networks. Others measure the **local** smoothness of the network in the vicinity of a particular input $x$. In theory, one could obtain a robustness guarantee via an upper bound on the local Lipschitz constant of the network (Hein & Andriushchenko, 2017), but computing this quantity is intractable for general neural networks. Instead, a panoply of practical solutions have been proposed in the literature (Wong & Kolter, 2018; Wang et al., 2018a,b; Raghunathan et al., 2018a,b; Wong et al., 2018; Dvijotham et al., 2018b,a; Croce et al., 2019; Gehr et al., 2018; Mirman et al., 2018; Singh et al., 2018; Gowal et al., 2018; Weng et al., 2018a; Zhang et al., 2018). Two themes stand out. Some approaches cast verification as an optimization problem and import tools such as relaxation and duality from the optimization literature to provide conservative guarantees (Wong & Kolter, 2018; Wang et al., 2018; Raghunathan et al., 2018a,b; Dvijotham et al., 2018b,a). Others step through the network layer by layer, maintaining at each layer an outer approximation of the set of activations reachable by a perturbed input (Mirman et al., 2018; Singh et al., 2018; Gowal et al., 2018; Weng et al., 2018a; Zhang et al., 2018). None of these local certification methods have been shown to be feasible on networks that are large and expressive enough to solve modern machine learning problems like the ImageNet classification task. Also, all method either assume specific network architectures (e.g. ReLU activations or a layered feedforward structure) or require extensive customization for new network architectures.

**Related work involving noise** Prior works have proposed using a network’s robustness to Gaussian noise as a proxy for its robustness to adversarial perturbations (Weng et al., 2018b; Ford et al., 2019), and have suggested that Gaussian data augmentation could supplement or replace adversarial training (Zantedeschi et al., 2017; Kannan et al., 2018). Smilkov et al. (2017) observed that averaging a classifier’s input gradients over Gaussian corruptions of an image yields very interpretable saliency maps. The robustness of neural networks to random noise has been analyzed both theoretically (Fawzi et al., 2016; Franceschi et al., 2018) and empirically (Dodge & Karam, 2017). Finally, Webb et al. (2019) proposed a statistical technique for estimating the noise robustness of a classifier more efficiently than naive Monte Carlo simulation; we did not use this technique since it appears to lack formal high-probability guarantees. While these works hypothesized relationships between a neural network’s robustness to random noise and the same network’s robustness to adversarial perturbations, randomized smoothing instead uses a classifier’s robustness to random noise to create a new classifier robust to adversarial perturbations.

**Randomized smoothing** Randomized smoothing has been studied previously for adversarial robustness. Several works (Liu et al., 2018; Cao & Gong, 2017) proposed similar techniques as heuristic defenses, but did not prove any guarantees. Lecuyer et al. (2019) used inequalities from the differential privacy literature to prove an $\ell_2$ and $\ell_1$ robustness guarantee for smoothing with Gaussian and Laplace noise, respectively. Subsequently, Li et al. (2018) used tools from information theory to prove a stronger $\ell_2$ robustness guarantee for Gaussian noise. However, all of these robustness guarantees are loose. In contrast, we prove a tight robustness guarantee in $\ell_2$ norm for randomized smoothing with Gaussian noise.

### 3. Randomized smoothing

Consider a classification problem from $\mathbb{R}^d$ to classes $\mathcal{Y}$. As discussed above, randomized smoothing is a method for constructing a new, “smoothed” classifier $g$ from an arbitrary base classifier $f$. When queried at $x$, the smoothed classifier $g$ returns whichever class the base classifier $f$ is most likely
to return when $x$ is perturbed by isotropic Gaussian noise:

$$g(x) = \arg\max_{c \in \mathcal{Y}} \mathbb{P}(f(x + \varepsilon) = c)$$  \hspace{1cm} (1)

where $\varepsilon \sim \mathcal{N}(0, \sigma^2 I)$

An equivalent definition is that $g(x)$ returns the class $c$ whose pre-image $\{x' \in \mathbb{R}^d : f(x') = c\}$ has the largest probability measure under the distribution $\mathcal{N}(x, \sigma^2 I)$. The noise level $\sigma$ is a hyperparameter of the smoothed classifier $g$ which controls a robustness/accuracy tradeoff; it does not change with the input $x$. We leave undefined the behavior of $g$ when the argmax is not unique.

We will first present our robustness guarantee for the smoothed classifier $g$. Then, since it is not possible to exactly evaluate the prediction of $g$ at $x$ or to certify the robustness of $g$ around $x$, we will give Monte Carlo algorithms for both tasks that succeed with arbitrarily high probability.

### 3.1. Robustness guarantee

Suppose that when the base classifier $f$ classifies $\mathcal{N}(x, \sigma^2 I)$, the most probable class $c_A$ is returned with probability $p_A$, and the “runner-up” class is returned with probability $p_B$. Our main result is that smoothed classifier $g$ is robust around $x$ within the $\ell_2$ radius $R = \frac{\sigma}{2} (\Phi^{-1}(p_A) - \Phi^{-1}(p_B))$, where $\Phi^{-1}$ is the inverse of the standard Gaussian CDF. This result also holds if we replace $p_A$ with a lower bound $\underline{p}_A$ and we replace $p_B$ with an upper bound $\overline{p}_B$.

**Theorem 1.** Let $f : \mathbb{R}^d \to \mathcal{Y}$ be any deterministic or random function, and let $\varepsilon \sim \mathcal{N}(0, \sigma^2 I)$. Let $g$ be defined as in (1). Suppose $c_A \in \mathcal{Y}$ and $p_A, \overline{p}_B \in [0, 1]$ satisfy:

$$\mathbb{P}(f(x + \varepsilon) = c_A) \geq p_A \geq \overline{p}_B \geq \max_{c \neq c_A} \mathbb{P}(f(x + \varepsilon) = c)$$  \hspace{1cm} (2)

Then $g(x + \delta) = c_A$ for all $||\delta||_2 < R$, where

$$R = \frac{\sigma}{2} (\Phi^{-1}(p_A) - \Phi^{-1}(\overline{p}_B))$$  \hspace{1cm} (3)

We now make several observations about Theorem 1:

- **Theorem 1** assumes nothing about $f$. This is crucial since it is unclear which well-behavedness assumptions, if any, are satisfied by modern deep architectures.

- The certified radius $R$ is large when: (1) the noise level $\sigma$ is high, (2) the probability of the top class $p_A$ is high, and (3) the probability of each other class is low.

- The certified radius $R$ goes to $\infty$ as $p_A \to 1$ and $\overline{p}_B \to 0$. This should sound reasonable: the Gaussian distribution is supported on all of $\mathbb{R}^d$, so the only way that $f(x + \varepsilon) = c_A$ with probability 1 is if $f = c_A$ almost everywhere.

Both Lecuyer et al. (2019) and Li et al. (2018) proved $\ell_2$ robustness guarantees for the same setting as Theorem 1, but with different, smaller expressions for the certified radius. However, our $\ell_2$ robustness guarantee is tight: if (2) is all that is known about $f$, then it is impossible to certify an $\ell_2$ ball with radius larger than $R$. In fact, it is impossible to certify any superset of the $\ell_2$ ball with radius $R$.

**Theorem 2.** Assume $p_A + \overline{p}_B \leq 1$. For any perturbation $\delta$ with $||\delta||_2 > R$, there exists a base classifier $f$ consistent with the class probabilities (2) for which $g(x + \delta) \neq c_A$.

Theorem 2 shows that Gaussian smoothing naturally induces $\ell_2$ robustness: if we make no assumptions on the base classifier beyond the class probabilities (2), then the set of perturbations to which a Gaussian-smoothed classifier is provably robust is exactly an $\ell_2$ ball.

The complete proofs of Theorems 1 and 2 are in Appendix A. We now sketch the proofs in the special case where there are only two classes.

**Theorem 1 (binary case).** Suppose $p_A \in (\frac{1}{2}, 1]$ satisfies $\mathbb{P}(f(x + \varepsilon) = c_A) \geq p_A$. Then $g(x + \delta) = c_A$ for all $||\delta||_2 < \sigma \Phi^{-1}(p_A)$.

**Proof sketch.** Fix a perturbation $\delta \in \mathbb{R}^d$. To guarantee that $g(x + \delta) = c_A$, we need to show that $f$ classifies the translated Gaussian $\mathcal{N}(x + \delta, \sigma^2 I)$ as $c_A$ with probability $\frac{\sigma}{2}$. However, all we know about $f$ is that $f$ classifies $\mathcal{N}(x, \sigma^2 I)$ as $c_A$ with probability $p_A$. This raises the question: out of all possible base classifiers $f$ which classify $\mathcal{N}(x, \sigma^2 I)$ as $c_A$ with probability $p_A$, which one $f^*$ classifies $\mathcal{N}(x + \delta, \sigma^2 I)$ as $c_A$ with the smallest probability? One can show using an argument similar to the Neyman-Pearson lemma (Neyman & Pearson, 1933) that this “worst-case” $f^*$ is a linear classifier whose decision boundary is normal to the perturbation $\delta$ (Figure 3):

$$f^*(x') = \begin{cases} c_A & \text{if } \delta^T (x' - x) \leq \sigma ||\delta||_2 \Phi^{-1}(p_A) \\ c_B & \text{otherwise} \end{cases}$$  \hspace{1cm} (4)

This “worst-case” $f^*$ classifies $\mathcal{N}(x + \delta, \sigma^2 I)$ as $c_A$ with probability $\Phi \left( \Phi^{-1}(p_A) - \frac{||\delta||_2}{\sigma} \right)$. Therefore, to ensure that even the “worst-case” $f^*$ classifies $\mathcal{N}(x + \delta, \sigma^2 I)$ as $c_A$ with probability $\frac{1}{2}$, we solve for those $\delta$ for which

$$\Phi \left( \Phi^{-1}(p_A) - \frac{||\delta||_2}{\sigma} \right) > \frac{1}{2}$$

which is equivalent to the condition $||\delta||_2 < \sigma \Phi^{-1}(p_A)$. 

Theorem 2 is a simple consequence: for any $\delta$ with $||\delta||_2 > R$, the base classifier $f^*$ defined in (4) is consistent with (2); yet if $f^*$ is the base classifier, then $g(x + \delta) = c_B$. 

Certified Adversarial Robustness via Randomized Smoothing
on the probability that $p$.

Figure 3. Illustration of $f^*$ in two dimensions. The concentric circles are the density contours of $N(x, \sigma^2 I)$ and $N(x + \delta, \sigma^2 I)$. Out of all base classifiers $f$ which classify $N(x, \sigma^2 I)$ as $c_A$ (blue) with probability $\geq p_A$, such as both classifiers depicted above, the “worst-case” $f^*$, which classifies $N(x + \delta, \sigma^2 I)$ as $c_A$ with minimal probability, is the classifier depicted on the right: a linear classifier with decision boundary normal to the perturbation $\delta$.

Figure 5 (left) plots our $\ell_2$ robustness guarantee against the guarantees derived in prior work. Observe that our $R$ is much larger than that of Lecuyer et al. (2019) and moderately larger than that of Li et al. (2018). Appendix I derives the other two guarantees using this paper’s notation.

Linear base classifier A two-class linear classifier $f(x) = \text{sign}(w^T x + b)$ is already certifiable: the distance from any input $x$ to the decision boundary is $|w^T x + b|/||w||$, and no perturbation $\delta$ with $\ell_2$ norm less than this distance can possibly change $f$’s prediction. In Appendix B we show that if $f$ is linear, then the smoothed classifier $g$ is identical to the base classifier $f$. Moreover, we show that our bound (3) will certify the true robust radius $|w^T x + b|/||w||$, rather than a smaller, overconservative radius. Therefore, when $f$ is linear, there always exists a perturbation $\delta$ just beyond the certified radius which changes $g$’s prediction.

Noise level can scale with image resolution Since our expression (3) for the certified radius does not depend explicitly on the data dimension $d$, one might worry that randomized smoothing is less effective for images of higher resolution — certifying a fixed $\ell_2$ radius is “less impressive” for, say, a $224 \times 224$ image than for a $56 \times 56$ image. However, as illustrated by Figure 4, images in higher resolution can tolerate higher levels $\sigma$ of isotropic Gaussian noise before their class-distinguishing content gets destroyed. As a consequence, in high resolution, smoothing can be performed with a larger $\sigma$, leading to larger certified radii. See Appendix G for a more rigorous version of this argument.

3.2. Practical algorithms

We now present practical Monte Carlo algorithms for evaluating $g(x)$ and certifying the robustness of $g$ around $x$.

Figure 4. Left to right: clean $56 \times 56$ image, clean $224 \times 224$ image, noisy $56 \times 56$ image ($\sigma = 0.5$), noisy $224 \times 224$ image ($\sigma = 0.5$).

3.2.1. Prediction

Evaluating the smoothed classifier’s prediction $g(x)$ requires identifying the class $c_A$ with maximal weight in the categorical distribution $f(x + \eps)$. The procedure described in pseudocode as PREDICT draws $n$ samples of $f(x + \eps)$ by running $n$ noise-corrupted copies of $x$ through the base classifier. Let $\hat{c}_A$ be the class which appeared the largest number of times. If $\hat{c}_A$ appeared much more often than any other class, then PREDICT returns $\hat{c}_A$. Otherwise, it abstains from making a prediction. We use the hypothesis test from Hung & Fithian (2019) to calibrate the abstention threshold so as to bound by $\alpha$ the probability of returning an incorrect answer. PREDICT satisfies the following guarantee:

**Proposition 1.** With probability at least $1 - \alpha$ over the randomness in PREDICT, PREDICT will either abstain or return $g(x)$. (Equivalently: the probability that PREDICT returns a class other than $g(x)$ is at most $\alpha$.)

The function SAMPLEUNDENOISE($f, x, \text{num}, \sigma$) in the pseudocode draws num samples of noise, $\eps_1 \ldots \eps_{\text{num}} \sim \mathcal{N}(0, \sigma^2 I)$, runs each $x + \eps_i$ through the base classifier $f$, and returns a vector of class counts. BINOMIALVALUE($n_A$, $n_A + n_B$, $p$) returns the $p$-value of the two-sided hypothesis test that $n_A \sim \text{Binomial}(n_A + n_B, p)$.

Even if the true smoothed classifier $g$ is robust at radius $R$, PREDICT will be vulnerable in a certain sense to adversarial perturbations with $\ell_2$ norm slightly less than $R$. By engineering a perturbation $\delta$ for which $f(x + \delta + \eps)$ puts mass just over $\frac{1}{2}$ on class $c_A$ and mass just under $\frac{1}{2}$ on class $c_B$, an adversary can force PREDICT to abstain at a high rate. If this scenario is of concern, a variant of Theorem 1 could be proved to certify a radius in which $\mathbb{P}(f(x + \delta + \eps) = c_A)$ is larger by some margin than $\max_{c \neq c_A} \mathbb{P}(f(x + \delta + \eps) = c)$.

3.2.2. Certification

Evaluating and certifying the robustness of $g$ around an input $x$ requires not only identifying the class $c_A$ with maximal weight in $f(x + \eps)$, but also estimating a lower bound $p_A$ on the probability that $f(x + \eps) = c_A$ and an upper bound $\overline{p}_A$ on the probability that $f(x + \eps)$ equals any other class. Doing all three of these at the same time in a statistically correct manner requires some care. One simple
Theorem 1 holds regardless of how the base classifier $f$ is trained. However, in order for $g$ to classify the labeled example $(x, c)$ correctly and robustly, $f$ needs to consistently classify $\mathcal{N}(x, \sigma^2 I)$ as $c$. In high dimension, the Gaussian distribution $\mathcal{N}(x, \sigma^2 I)$ places almost no mass near its mode $x$. As a consequence, when $\sigma$ is moderately high, the distribution of natural images has virtually disjoint support from the distribution of natural images corrupted by $\mathcal{N}(0, \sigma^4 I)$; see Figure 2 for a visual demonstration. Therefore, if the base classifier $f$ is trained via standard supervised learning on the data distribution, it will see no noisy images during training, and hence will not necessarily learn to classify $\mathcal{N}(x, \sigma^2 I)$ with $x$’s true label. Indeed, we observed empirically that when neural network base classifiers are trained on noiseless data, they cannot recognize noisy images.

Therefore, in this paper we follow Lecuyer et al. (2019) and train the base classifier with Gaussian data augmentation at variance $\sigma^2$. A justification for this procedure is provided in Appendix F. However, we suspect that there may be room to improve upon this training scheme, perhaps by training the base classifier so as to maximize the smoothed classifier’s certified accuracy at some tunable radius $r$.

4. Experiments

In adversarially robust classification, one metric of interest is the certified test set accuracy at radius $r$, defined as the fraction of the test set which $g$ classifies correctly with a prediction that is certifiably robust within an $\ell_2$ ball of radius $r$. However, if $g$ is a randomized smoothing classifier, computing this quantity exactly is not possible, so we instead report the approximate certified test set accuracy, defined as the fraction of the test set which CERTIFY classifies correctly (without abstaining) and certifies robust with a radius $R \geq r$. Appendix D shows how to convert the approximate certified accuracy into a lower bound on the true certified accuracy that holds with high probability over the randomness in CERTIFY. However Appendix H.2 demonstrates that when $\alpha$ is small, the difference between these two quantities is negligible. Therefore, in our experiments we omit the step for simplicity and report approximate certified accuracies.
In all experiments, unless otherwise stated, we ran CERTIFY with $\alpha = 0.001$, so there was at most a 0.1% chance that CERTIFY returned a radius in which $g$ was not truly robust. Unless otherwise stated, when running CERTIFY we used $n_0 = 100$ Monte Carlo samples for selection and $n = 100,000$ samples for estimation.

In the figures above that plot certified accuracy as a function of radius $r$, the certified accuracy always decreases gradually with $r$ until reaching some point where it plummets to zero. This drop occurs because for each noise level $\sigma$ and number of samples $n$, there is a hard upper limit to the radius we can certify with high probability, achieved when all $n$ samples are classified by $f$ as the same class.

**ImageNet and CIFAR-10 results** We applied randomized smoothing to CIFAR-10 (Krizhevsky, 2009) and ImageNet (Deng et al., 2009). On each dataset we trained several smoothed classifiers, each with a different $\sigma$. On CIFAR-10 our base classifier was a 110-layer residual network; certifying each example took 15 seconds on an NVIDIA RTX 2080 Ti. On ImageNet our base classifier was a ResNet-50; certifying each example took 110 seconds. We also trained a neural network with the base classifier’s architecture on clean data, and subjected it to a DeepFool $\ell_2$ adversarial attack (Moosavi-Dezfooli et al., 2016), in order to obtain an empirical upper bound on its robust accuracy. We certified the full CIFAR-10 test set and a subsample of 500 examples from the ImageNet test set.

**Comparison to baselines** We compared randomized smoothing to three baseline approaches for certified $\ell_2$ robustness: the duality approach from Wong et al. (2018), the Lipschitz approach from Tsuzuku et al. (2018), and the

---

**Figure 6.** Approximate certified accuracy attained by randomized smoothing on CIFAR-10 (top) and ImageNet (bottom). The hyper-parameter $\sigma$ controls a robustness/accuracy tradeoff. The dashed black line is an upper bound on the empirical robust accuracy of an undefended classifier with the base classifier’s architecture.

**Figure 7.** Comparison between randomized smoothing and Wong et al. (2018). Each green line is a small resnet classifier trained and certified using the method of Wong et al. (2018) with a different setting of its hyperparameter $\epsilon$. The purple line is our method using the same small resnet architecture as the base classifier; the blue line is our method with a larger neural network as the base classifier. Wong et al. (2018) gives deterministic robustness guarantees, whereas smoothing gives high-probability guarantees; therefore, we plot here the certified accuracy of Wong et al. (2018) against the “approximate” certified accuracy of smoothing.
Figure 8. Experiments with randomized smoothing on ImageNet with $\sigma = 0.25$. **Left**: certified accuracies obtained using our Theorem 1 versus those obtained using the robustness guarantees derived in prior work. **Middle**: projections for the certified accuracy if the number of samples $n$ used by CERTIFY had been larger or smaller. **Right**: certified accuracy as the failure probability $\alpha$ of CERTIFY is varied.

approach from Weng et al. (2018a); Zhang et al. (2018). The strongest baseline was Wong et al. (2018); we defer the comparison to the other two baselines to Appendix H.

In Figure 7, we compare the largest publicly released model from Wong et al. (2018), a small resnet, to two randomized smoothing classifiers: one which used the same small resnet architecture for its base classifier, and one which used a larger 110-layer resnet for its base classifier. First, observe that smoothing with the large 110-layer resnet substantially outperforms the baseline (across all hyperparameter settings) at all radii. Second, observe that smoothing with the small resnet also outperformed the method of Wong et al. (2018) at all but the smallest radii. We attribute this latter result to the fact that neural networks trained using the method of Wong et al. (2018) are “typically overregularized to the point that many filters/weights become identically zero,” per that paper. In contrast, the base classifier in randomized smoothing is a fully expressive neural network.

Prediction It is computationally expensive to certify the robustness of $g$ around a point $x$, since the value of $n$ in CERTIFY must be very large. However, it is far cheaper to evaluate $g$ at $x$ using PREDICT, since $n$ can be small. For example, when we ran PREDICT on ImageNet ($\sigma = 0.25$) using $n = 100$, making each prediction only took 0.15 seconds, and we attained a top-1 test accuracy of 65% (Appendix E).

As discussed earlier, an adversary can potentially force PREDICT to abstain with high probability. However, it is relatively rare for PREDICT to abstain on the actual data distribution. On ImageNet ($\sigma = 0.25$), PREDICT with failure probability $\alpha = 0.001$ abstained 12% of the time when $n = 100$, 4% when $n = 1000$, and 1% when $n = 10,000$.

Empirical tightness of bound When $f$ is linear, the bound in Theorem 1 is tight, in that there always exists a class-changing perturbation just beyond the certified radius. Since deep neural networks are not linear, we empirically assessed the tightness of our bound by subjecting an ImageNet randomized smoothing classifier ($\sigma = 0.25$) to a projected gradient descent-style adversarial attack. For each example, we ran CERTIFY with $\alpha = 0.01$, and, if the example was correctly classified and certified robust at radius $R$, we tried finding an adversarial example for $g$ within radius $1.5R$ and within radius $2R$. We succeeded 17% of the time at radius $1.5R$ and 53% of the time at radius $2R$. See Appendix J.3 for more details on the attack.

5. Conclusion

Theorem 2 establishes that smoothing with Gaussian noise naturally confers adversarial robustness in $\ell_2$ norm: if we have no knowledge about the base classifier beyond the distribution of $f(x + \varepsilon)$, then the set of perturbations to which the smoothed classifier is provably robust is precisely an $\ell_2$ ball. We suspect that smoothing with other noise distributions may lead to similarly natural robustness guarantees for otherperturbation sets such as general $\ell_p$ norm balls.

Our strong empirical results suggest that randomized smoothing is a promising direction for future research into adversarially robust classification. Most empirical approaches (except PGD adversarial training) have been “broken,” and provable approaches based on certifying neural network classifiers have not been shown to scale to networks of modern size. It seems to be computationally infeasible to reason in any sophisticated way about the decision boundaries of a large, expressive neural network. Randomized smoothing circumvents this problem: the smoothed classifier is not itself a neural network, though it leverages the discriminative ability of a neural network base classifier. To make the smoothed classifier robust, one need simply make the base classifier classify well under noise. In this way, randomized smoothing reduces the unsolved problem of adversarially robust classification to the comparably solved domain of supervised learning.
6. Acknowledgements

We thank Mateusz Kwaśnicki for help with Lemma 4 in the appendix, Aaditya Ramdas for pointing us toward the work of Hung & Fithian (2019), and Siva Balakrishnan for helpful discussions regarding the confidence interval in Appendix D. We thank Tolani Olarinre, Adarsh Prasad, Ben Cousins, and Ramon Van Handel for useful conversations. Finally, we are very grateful to Vaishnavh Nagarajan, Arun Sai Suggala, Shaojie Bai, Mikhail Khodak, Han Zhao, and Zachary Lipton for reviewing drafts of this work. Jeremy Cohen is supported by a grant from the Bosch Center for AI.

References


Certified Adversarial Robustness via Randomized Smoothing


Neyman, J. and Pearson, E. S. On the problem of the most efficient tests of statistical hypotheses. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 231:289–337, 1933.


