Distributional Adversarial Loss

Saba Ahmadi* Toyota Technological Institue at Chicago

Siddharth Bhandari* Toyota Technological Institue at Chicago

Avrim Blum* Toyota Technological Institue at Chicago

Chen Dan* Toyota Technological Institue at Chicago

Prabhav Jain* Toyota Technological Institue at Chicago

Abstract

We initiate the study of a new notion of adversarial loss which we call *distributional adversarial loss*. In this notion, we assume for each original example, the allowed adversarial perturbation set is a family of *distributions*, and the adversarial loss over each example is the maximum loss over all the associated distributions. The goal is to minimize the overall adversarial loss. We show sample complexity bounds in the PAC-learning setting for our notion of adversarial loss. Our notion of adversarial loss contrasts the prior work on robust learning that considers a set of points, not distributions, as the perturbation set of each clean example. As an application of our approach, we show how to unify the two lines of work on randomized smoothing and robust learning in the PAC-learning setting and derive sample complexity bounds for randomized smoothing methods.

Furthermore, we investigate the role of randomness in achieving robustness against adversarial attacks. We show a general derandomization technique that preserves the extent of a randomized classifier's robustness against adversarial attacks and show its effectiveness empirically.

1 Introduction and Related Work

Recent research extensively explores the development of robust predictors against adversarial perturbations, revealing the susceptibility of deep neural networks to imperceptible adversarial noise Biggio et al. [2013], Szegedy et al. [2013], Goodfellow et al. [2015]. Adversarial perturbations involve introducing limited noise δ to an image x (or more generally, $x' \in \mathcal{A}(x)$ for a perturbation set $\mathcal{A}(x)$), resulting in visually indistinguishable yet misclassified instances. This phenomenon poses significant threats to real-world applications such as self-driving cars Cao et al. [2021] and healthcare Finlayson et al. [2019]. To bolster classifier resilience against these perturbations, various empirical defenses have been proposed, however, these methods often overfit and still exhibit vulnerability to meticulously crafted adversaries on test points [Carlini and Wagner, 2017, Athalye and Carlini, 2018].

Subsequently, techniques for *certifiable robustness* have been introduced. These methods ensure that for any given input x, whether it is clean or perturbed, a radius ρ can be determined such that all inputs x' within the distance ρ from the original input x are guaranteed to receive the same label as x. *Randomized smoothing* methods [Cohen et al., 2019, Salman et al., 2019, Zhai et al., 2020, Mohapatra et al., 2020] have been proposed as certifiable robustness techniques that scale to large-scale datasets such as ImageNet. In randomized smoothing, given a possibly perturbed input, the final classification is provided by taking the majority vote over the Gaussian-smoothed perturbations of the input, or other suitable smoothing perturbations. The principle here is that the added noise helps to drown out the adversarial perturbation present in the input and counteract an adversary's power.

In another line of work, the sample complexity of robust learning in the PAC-learning setting has been studied, e.g. Cullina et al. [2018], Attias et al. [2022], Montasser et al. [2019], where the goal is to learn a predictor having small robust loss that is defined as $\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\max_{z\in\mathcal{A}(x)} \mathbb{1}\left[h(z)\neq y\right]\right]$. That is, for each non-adversarial example sampled from an underlying distribution $\mathcal{D}$, the adversarial loss on (x, y) is defined as the maximum loss over all its perturbations in $\mathcal{A}(x)$.

Motivated by these two lines of work, we propose a new notion of adversarial loss which we call *distributional adversarial loss*. Here, for each example x, the perturbation set $\mathcal{U}(x)$ is a family of *distributions* instead of

Proceedings of the 28th International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258.

a set of points. A perturbation x' of x can be sampled from any of the distributions $\mathcal{U}(x)$. $\mathcal{U}(x)$ corresponds roughly to the set of distributions that arise when the adversary chooses a perturbed version of x, say x', and x' is smoothed by adding noise to it. Our notion contrasts the prior work that considers a set of points, not distributions, as the perturbation set of each clean example. Given that, the distributional adversarial loss is defined as $\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\max_{u\in\mathcal{U}(x)}\left[\mathbb{E}_{z\sim u}\,\mathbb{1}\left[h(z)\neq y\right]\right]\right]$. That is, for each non-adversarial example sampled from an underlying distribution $\mathcal{D}$, the adversarial loss on (x, y) is defined as the maximum expected loss over all the distributions in its perturbation set $\mathcal{U}(x)$. The population loss is the expected adversarial loss over all the samples coming from the underlying distribution $\mathcal{D}$. We show PAC-learning results for this notion of adversarial loss.

We expand on two important and desirable properties of optimizing our notion of distributional adversarial loss. First, using our framework, we can derive sample complexity bounds for randomized smoothing methods (Section 2.5.2). Another benefit of our approach is that in some settings, it happens that for each input x, any adversarial choice $x' \in \mathcal{A}(x)$ is covered by some distribution $u \in \mathcal{U}(x)$[1]. Therefore, by having a low distributional adversarial loss on u, the classifier would also have a good performance on x'. In some settings, including randomized smoothing, we can consider a distribution set $\mathcal{U}(x)$ such that $\mathcal{U}(x)$ is much smaller than $\mathcal{A}(x)$. As a result, instead of considering an *unbounded* number of adversary's actions, it suffices to consider a *bounded* number of distributions that cover the adversary's actions. Another way of viewing this is that now *the same adversary with the same power* has a smaller set of *effective* actions that we need to worry about, a bounded number of distributions that cover an unbounded number of perturbation points.

Derandomization: Another natural point of inquiry about the robust classifiers (certifiable or not) built using the above approaches is whether they can be made deterministic, i.e., no randomness is required during inference time. This question has been answered in the affirmative in some cases by Levine and Feizi [2021]. There are a few advantages to having a deterministic classifier. In pivotal decisions, we aim for the classifier's output to be deterministic, ensuring consistent labeling for a given input (likewise for the certification radius). Additionally, derandomization has the potential to enhance the classifier's robustness against adversarial attacks (see Section 3 for more discussion). In this work, we show a generic procedure to derandomize any robust classifier, which preserves the extent of its robustness and certification.

[1]By covering, we mean that u is close in some notion of distance, e.g. total variation distance, to a distribution supported on a ball around x'. The radius of the ball depends on the adversary's power. More details in Section 2.5.2.

Our Contributions.

- We show that bounded VC-dimension is sufficient for distributionally adversarial PAC-learning with a proper learning rule. [Section 2, Theorems 2.1 and 2.3]
- One important application of our framework is to derive sample complexity bounds for randomized smoothing methods, unifying the line of work on robust learning in PAC-learning setting and randomized smoothing methods. [Section 2.5.2]
- We show a general derandomization technique that preserves the extent of a randomized classifier's robustness and certification. Additionally, our experimental findings suggest that this approach has the potential to enhance the robust accuracy of the initial classifier.

All the missing proofs are in the Appendix.

Further related work. Prior work has studied the sample complexity of robust learning in the PAC-learning setting. Among others, Cullina et al. [2018] derived sample complexity results for this problem through a notion called adversarial VC-dimension. Attias et al. [2022]'s sample complexity depends on the number of perturbations allowed for each instance. Montasser et al. [2019]'s upper bound depends on the VC and dual VC dimension of a hypothesis class, and they use an improper learning rule. Montasser et al. [2022] demonstrated a characterization of robust learning based on the one-inclusion graph of Haussler et al. [1994] adapted to robust learning. In our work unlike the prior work, we model the perturbation set of points as a set of distributions instead of a set of points and connect our notion to randomized smoothing methods.

2 Distributional Adversarial Loss

2.1 Problem Setup

Loss Function We are given an instance space $\mathcal{X}$ and label space $\mathcal{Y} = \{-1, +1\}$ and a distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$. For each unperturbed (clean) input $(x, y) \sim \mathcal{D}$, i.e. input x with label y, the perturbation set $\mathcal{U}(x)$ is a family of distributions, where a perturbation z of (x, y) is sampled from an adversarially chosen distribution $u \in \mathcal{U}(x)$. Given that, the distributional adversarial loss of a

classifier h is defined as:

$$\mathsf{DAL}_{\mathcal{D}}(h) = \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \left[\max_{u\in\mathcal{U}(x)} \left[\mathop{\mathbb{E}}_{z\sim u} \mathbb{1}\left[h(z) \neq y\right] \right] \right] \quad (1)$$

That is, for each non-adversarial example sampled from an underlying distribution $\mathcal{D}$, the robust loss on (x, y) is defined as the maximum expected loss over all the distributions in its perturbation set $\mathcal{U}(x)$. The objective is to minimize the expected robust loss over all the samples coming from the underlying distribution $\mathcal{D}$. First, we consider a setting where for each example (x, y), the size of the perturbation set is bounded, i.e. $|\mathcal{U}(x)| \leq k$ for some value $k > 0$. Next, we consider two further extensions.

In the first extension in Section 2.5, we investigate a setting where $\mathcal{U}(x)$ is potentially unbounded, however, there exists a set of representative distributions $\mathcal{R}(x) \subseteq \mathcal{U}(x)$ where $|\mathcal{R}(x)| \leq k$ and for each distribution $u \in \mathcal{U}(x) \setminus \mathcal{R}(x)$, there exists a representative distribution $r \in \mathcal{R}(x)$ where the total variation distance between u and r is bounded. Consequently, in Section 2.5.2, we demonstrate that this extension captures the sample complexity of randomized smoothing methods.

Furthermore, in Section 2.5.3, we consider a scenario where the number of distribution perturbations is unbounded, however, there exists a set of *representative* distributions of size at most k such that each distribution $u \in \mathcal{U}(x)$, is completely covered by $\mathcal{R}(x)$, i.e. the probability density function of u is point-wise bounded by the maximum of probability density functions of the distributions in $\mathcal{R}(x)$.

PAC Learning of Distributional Adversarial Loss: We study the *sample complexity* for *PAC-learning of distributional adversarial loss* in the realizable and agnostic settings. Given a hypothesis class $\mathcal{H} \subseteq \mathcal{Y}^{\mathcal{X}}$, our goal is to find a learning rule $\mathcal{L}$ such that for any distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$ finds a classifer $h \in \mathcal{H}$ that competes with the predictor $h^* \in \mathcal{H}$ where $h^* = \text{argmin}_{h\in\mathcal{H}} \mathsf{DAL}_{\mathcal{D}}(h)$, using a number of samples that is independent of the underlying distribution $\mathcal{D}$. In the following, we formally define the notion of distributionally adversarial PAC learning in realizable and agnostic settings.

Definition 2.1 (Agnostic Distributionally Adversarial PAC learning)**.** *A hypothesis class $\mathcal{H}$ is agnostic distributionally adversarial PAC-learnable if there exists functions $n_{\mathcal{H}} : (0, 1)^2 \to N$ and $m : (0, 1) \to N$ and a learning algorithm $\mathcal{L}$ with the following property: For every $\varepsilon, \delta \in (0, 1)$, and for every distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, when running the learning algorithm on a set $S = S_c \cup S_p$ where the set of clean (unperturbed) examples S_c is consisting of $n \geq n_{\mathcal{H}}(\varepsilon, \delta)$ i.i.d. examples sampled from $\mathcal{D}$ and for each example $(x, y) \in S_c$, $m \geq m(\varepsilon)$ perturbations sampled from each $u \in \mathcal{U}(x)$ and added to the perturbations set S_p, the algorithm returns $h \in \mathcal{H}$ such that, with probability of at least $1 - \delta$,*

$$\mathsf{DAL}_{\mathcal{D}}(h) \leq \min_{h^*\in\mathcal{H}} \mathsf{DAL}_{\mathcal{D}}(h^*) + \varepsilon$$

Similarly, we can define Realizable Distributionally Adversarial PAC learning where the goal is to find a predictor $h \in \mathcal{H}$ such that, with probability of at least $1 - \delta$, $\mathsf{DAL}_{\mathcal{D}}(h) \leq \varepsilon$.

We show that bounded VC-dimension is sufficient for distributionally adversarial PAC-learning. Theorems 2.1 and 2.3 prove this in the realizable and agnostic case respectively.

2.2 Learning Algorithm:

In this section, we describe our learning algorithm. Usually, in the PAC-learning setting, the idea is to sample enough number of examples from the underlying distribution, and find the best predictor h^* on the sampled dataset. Furthermore, using uniform convergence guarantees, it can be argued that assuming enough number of examples are sampled from the underlying distribution, the performance of h^* on the sampled dataset and the true distribution are close.

However, for our problem, we need a two-layer sampling procedure. This happens since in the definition of distributional adversarial loss (Equation (1)), there are two expectations, one for the underlying distribution $\mathcal{D}$ over the clean (unperturbed) examples, and the second one for each perturbation distribution u of an example coming from $\mathcal{D}$. As a result, in order to derive generalization guarantees, first, we need to sample enough number of examples from the underlying distribution $\mathcal{D}$. Furthermore, since each sampled example (x, y) has a set of distributions $\mathcal{U}(x)$ as its perturbation set, for each distribution $u \in \mathcal{U}(x)$, we need to draw a number of samples from u in order to argue about performance on each perturbation distribution u in addition to the performance on the underlying distribution $\mathcal{D}$.

Our training procedure is as follows. First, we draw a sample set S_c of clean (unperturbed) examples of size $n \geq n_{\mathcal{H}}(\varepsilon, \delta)$ i.i.d from $\mathcal{D}$. Then, for each example $(x, y) \in S$, we draw $m \geq m(\varepsilon)$ samples i.i.d. from each of the distributions $u \in \mathcal{U}(x)$ (or from each $u \in \mathcal{R}(x)$ when $\mathcal{U}(x)$ is unbounded) and add to the perturbations set S_p and let the training set be $S = S_c \cup S_p$. Therefore, $|S| \leq n \cdot m \cdot k + n$. For the training, we assume having access to an oracle DALERM that minimizes *empirical*

distributional adversarial loss:

$$\hat{h} \in \mathsf{DALERM}_{\mathcal{H}}(S) = \operatorname*{argmin}_{h \in \mathcal{H}} \mathsf{DAL}_S(h)$$

where the empirical distributional adversarial loss is defined as follows:

$$\mathsf{DAL}_S(h) = \frac{1}{n} \sum_{(x,y) \in S_c} \left[\max_{u \in \mathcal{U}(x)} \left[\frac{1}{m} \sum_{z \in \mathcal{U}(x) \cap S_p} \mathbb{1}\left[h(z) \neq y\right] \right] \right]$$

Now we are ready to demonstrate our generalization guarantees for distributional adversarial loss.

2.3 Realizable Distributionally Adversarial PAC Learning:

In this section, we focus on the setting where for each example (x, y), the size of their perturbation set is bounded, i.e. $|\mathcal{U}(x)| \leq k$. Theorem 2.1 exhibits sample complexity bounds in the realizable setting. As mentioned before, we need a two-layer sampling procedure in our setting. Consequently, Theorem 2.1 bounds the number of samples needed from the underlying distribution $\mathcal{D}$ over unperturbed examples, and the number of perturbations needed to be sampled from each distribution u in the perturbation set of a sampled example x.

Theorem 2.1 (VC-dimension sample bound in the realizable distributionally adversarial case)**.** *For any class $\mathcal{H}$ and distribution $\mathcal{D}$, a training sample S_c of size $n = \mathcal{O}\Big(\frac{1}{\varepsilon}[VCdim(\mathcal{H}) \log(\frac{mk}{\varepsilon}) + \log(\frac{1}{\delta}) + \frac{1}{\varepsilon}]\Big)$ where for each $(x, y) \in S_c$, $m = \Omega(1/\varepsilon^2 \log(1/\varepsilon))$ perturbations are sampled from each of the distributions $u \in \mathcal{U}(x)$. Let S_p denote the set of all perturbations, then $S = S_c \cup S_p$. Given sample set S, with probability $\geq 1-\delta$, every $h \in \mathcal{H}$ with $\mathsf{DAL}_{\mathcal{D}}(h) \geq \varepsilon$ has $\mathsf{DAL}_S(h) > 0$ (equivalently, every $h \in \mathcal{H}$ with $\mathsf{DAL}_S(h) = 0$ has $\mathsf{DAL}_{\mathcal{D}}(h) < \varepsilon$).*

In order to prove Theorem 2.1, first we show Lemma 2.1 holds, which states the following: Consider drawing a set S of n examples from $\mathcal{D}$ where for each example $(x, y) \in S$, m perturbations are drawn from each $u \in \mathcal{U}(x)$ and are added to S. Let A denote the event that there exists $h \in \mathcal{H}$ with zero empirical distributional adversarial error on S but true distributional adversarial error at least ε. Now draw a *fresh* test set S' of n examples from $\mathcal{D}$ where for each example $(x, y) \in S'$, m perturbations are drawn from each distribution $u \in \mathcal{U}(x)$ and are added to S'. Let B denote the event that there exists $h \in \mathcal{H}$ with zero empirical distributional adversarial loss on S but an empirical distributional adversarial loss at least $\varepsilon/2$ on S'. We prove that $\Pr(B) \geq (2/5)\Pr(A)$.

The purpose of this lemma is to show that we can argue about the error on a fresh test set instead of the true error on the underlying distribution. Later on, in Theorem 2.2, we show that for large enough training and test sets, the error values on the training and test sets are close. This implies that the training error is also close to the true error on the underlying distribution, hence it cannot be the case that $\mathsf{DAL}_S(h) = 0$ but $\mathsf{DAL}_{\mathcal{D}}(h) > \varepsilon$, and it proves Theorem 2.1.

Lemma 2.1. *Let $\mathcal{H}$ be a concept class over a domain $\mathcal{X}$. Let S_c and S_c' be sets of n clean (unperturbed) elements drawn from some distribution $\mathcal{D}$ over $\mathcal{X}$, where $n = \Omega(1/\varepsilon^2)$. For each $(x, y) \in S_c$, $m = \Omega(1/\varepsilon^2 \log(1/\varepsilon))$ perturbations sampled from each $u \in \mathcal{U}(x)$ are added to a set S_p and finally $S = S_c \cup S_p$. Similarly, S_c' is augmented to get S'. Let A denote the event that there exists $h \in \mathcal{H}$ with zero empirical distributional adversarial error on S but true distributional adversarial error $\geq \varepsilon$:*

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \left[\max_{u \in \mathcal{U}(x)} \left[\mathop{\mathbb{E}}_{z \sim u} \mathbb{1}\left[h(z) \neq y\right] \right] \right] \geq \varepsilon,$$

$$\frac{1}{n} \sum_{(x,y) \in S_c} \left[\max_{u \in \mathcal{U}(x)} \left[\frac{1}{m} \sum_{z \in \mathcal{U}(x) \cap S_p} \mathbb{1}\left[h(z) \neq y\right] \right] \right] = 0$$

Let B denote the event that there exists $h \in \mathcal{H}$ with zero distributional adversarial loss on S but distributional adversarial loss $\geq \varepsilon/2$ on S':

$$\frac{1}{n} \sum_{(x,y) \in S_c'} \left[\max_{u \in \mathcal{U}(x)} \left[\frac{1}{m} \sum_{z \in \mathcal{U}(x) \cap S_p'} \mathbb{1}\left[h(z) \neq y\right] \right] \right] \geq \varepsilon/2,$$

$$\frac{1}{n} \sum_{(x,y) \in S_c} \left[\max_{u \in \mathcal{U}(x)} \left[\frac{1}{m} \sum_{z \in \mathcal{U}(x) \cap S_p} \mathbb{1}\left[h(z) \neq y\right] \right] \right] = 0$$

Then $\Pr(B) \geq (2/5)\Pr(A)$.

Next, we prove Theorem 2.2. Proof of this theorem is similar to the original double-sampling trick by Vapnik and Chervonenkis [1971], Blumer et al. [1989] for showing sample complexity of PAC-learning. However, here, we also need to argue about the perturbations of the clean examples in the dataset. The idea is first to use the application of Lemma 2.1 to argue about the distributional adversarial loss on the test data instead of population distributional adversarial loss. Furthermore, for large enough training and test data, when sampled from the same distribution, it cannot be the case that the training distributional adversarial loss is low but the test distributional adversarial loss is large.

Theorem 2.2. *For any class $\mathcal{H}$ and distribution $\mathcal{D}$, a training sample S_c of size $n \geq \frac{2}{\varepsilon}[\log((5/2)\mathcal{H}[2n \cdot m \cdot k]) + \log(\frac{1}{\delta}) + \frac{7}{\varepsilon}]$ where for each $(x, y) \in S_c$, $m = \Omega(1/\varepsilon^2 \log(1/\varepsilon))$ perturbations are sampled from each of the distributions $u \in \mathcal{U}(x)$. Let S_p denote*

the set of all perturbations, then $S = S_c \cup S_p$. Let $k = \max_{x \in \mathcal{X}} |\mathcal{U}(x)|$. Given sample set S, with probability $\geq 1 - \delta$, every $h \in \mathcal{H}$ with $\mathsf{DAL}_{\mathcal{D}}(h) \geq \varepsilon$ has $\mathsf{DAL}_S(h) > 0$ (equivalently, every $h \in \mathcal{H}$ with $\mathsf{DAL}_S(h) = 0$ has $\mathsf{DAL}_{\mathcal{D}}(h) < \varepsilon$). Here, $\mathcal{H}[.]$ is the growth function of $\mathcal{H}$.

Putting together Theorem 2.2 and Lemma 2.1 and applying Sauer's Lemma proves Theorem 2.1. Details are deferred to the Appendix.

2.4 Agnostic Distributionally Adversarial PAC Learning

In this section, we present our sample complexity bounds in the agnostic setting where for each clean example, their perturbation set has size at most k. In this setting, Theorem 2.3 gives the sample complexity bound for distributionally adversarial PAC-learning. The idea to prove Theorem 2.3, is similar to Theorem 2.1. First, we exhibit that it suffices to argue about the distributional adversarial loss on a fresh test data instead of the population distributional adversarial loss. Furthermore, when the test and training data are coming from the same distribution, it cannot be the case that the training distributional adversarial loss is low but the test distributional adversarial loss is large. Finally, by the application of Sauer's lemma the proof is complete. Details are deferred to Appendix.

Theorem 2.3 (VC-dimension sample bound in the agnostic case)**.** *For any class $\mathcal{H}$ and distribution $\mathcal{D}$, a training sample S_c of size $n = \mathcal{O}\Big(\frac{1}{\varepsilon^2}[VCdim(\mathcal{H}) \log(\frac{mk}{\varepsilon}) + \log(\frac{1}{\delta})]\Big)$ where for each $(x, y) \in S_c$, $m = \Omega(1/\varepsilon^2 \log(1/\varepsilon))$ perturbations are sampled from each of the distributions $u \in \mathcal{U}(x)$. Let S_p denote the set of all perturbations, then $S = S_c \cup S_p$. Given sample set S, with probability $\geq 1 - \delta$, for every $h \in \mathcal{H}$, $|\mathsf{DAL}_{\mathcal{D}}(h) - \mathsf{DAL}_S(h)| \leq \varepsilon$.*

2.5 Extension to an arbitrary number of distributions

2.5.1 Model I

In this section, we consider the scenario where the number of distribution perturbations is unbounded, however, there exists a set $\mathcal{R}(x)$ of *representative* distributions of size at most k such that for each distribution $u \in \mathcal{U}(x)$, there exists a distribution $u_0 \in \mathcal{R}(x)$ such that the variation distance between u and u_0 is bounded (at most ε'). We show that using a similar learning algorithm that was used in the previous section, with the main difference that after drawing $n \geq n_{\mathcal{H}}(\varepsilon, \delta)$ i.i.d from $\mathcal{D}$ and adding to the training set, for each clean example $(x, y) \in S$, the perturbations are drawn from the representative distribution sets $\mathcal{R}(x)$ instead of the true distribution set $\mathcal{U}(x)$. In Theorem 2.4, we prove that training a predictor that minimizes the empirical distributional adversarial loss on S, will also minimize the population distributional adversarial loss with respect to the true perturbation sets $\mathcal{U}(.)$. A similar sample complexity bound holds in the agnostic setting (Theorem C.2). We show the connection of this model to randomized smoothing methods in Section 2.5.2.

Theorem 2.4 (realizable case)**.** *For any class $\mathcal{H}$ and distribution $\mathcal{D}$, a training sample S_c of size $n = \mathcal{O}\Big(\frac{1}{\varepsilon}[VCdim(\mathcal{H}) \log(\frac{mk}{\varepsilon}) + \log(\frac{1}{\delta}) + \frac{1}{\varepsilon}]\Big)$, where for each $(x, y) \in S_c$, $m = \Omega(1/\varepsilon^2 \log(1/\varepsilon))$ perturbations are sampled from each of the distributions $u \in \mathcal{R}(x)$. Let S_p denote the set of all perturbations, then $S = S_c \cup S_p$. Given sample set S, with probability $\geq 1 - \delta$, every $h \in \mathcal{H}$ with $\mathsf{DAL}_S(h) = 0$ has $\mathsf{DAL}_{\mathcal{D}}(h) < \varepsilon + \varepsilon'$. ε' is an upper bound on the total variation distance between each distribution $u \in \mathcal{U}(x)$ and the closest representative distribution $u_0 \in \mathcal{R}(x)$.*

2.5.2 Connection to Randomized Smoothing

Randomized smoothing constructs a new smoothed classifier g from a base classifier f. Given an input x, classifier g returns the class that the base classifier f is most likely to return given x is perturbed using a Gaussian noise:

$$g(x) = \underset{c \in \mathcal{Y}}{\operatorname{argmax}} \Pr(f(x + \eta) = c)$$

where $\eta \sim \mathcal{N}(0, \sigma^2 I)$. Cohen et al. [2019] exhibit certified robustness guarantees that a smoothed classifier g is robust around a clean input x within certain ℓ_2 radius. They show 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, then $g(x + \gamma) = c_A$ for all $\|\gamma\|_2 < R$, where $R = \sigma/2(\Phi^{-1}(p_A) - \Phi^{-1}(p_B))$ and Φ^{-1} is the inverse of the standard Gaussian CDF.

Our result is orthogonal to the guarantee exhibited by Cohen et al. [2019]. In their framework, the goal is that the smoothed classifier outputs the same prediction under perturbation as the base classifier, whether it is a correct prediction or not. However, in our framework, the goal is to derive sample complexity results such that assuming the unperturbed points are coming from a distribution, we train a classifier f such that after adding Gaussian noise to a clean input, it still predicts the correct label with high probability. Furthermore, our guarantees extend to the smoothed classifier g.

In order to exhibit sample complexity guarantees for randomized smoothing, we use Theorems 2.4 and C.2 where $k = 1$. For each example $(x, y) \sim \mathcal{D}$, the representative

distribution set $\mathcal{R}(x)$ consists of one single distribution that is a Gaussian around x, i.e. $\mathcal{N}(x, \sigma^2 I)$. During the test time, each example x can be perturbed by adding a limited perturbation γ to get x'. Given x' in the test time, the classifier g adds a Gaussian noise $\eta \sim \mathcal{N}(0, \sigma^2 I)$ to x' and outputs $g(x') = \text{argmax}_{c \in \mathcal{Y}} \Pr(f(x' + \eta) = c)$. Considering our model, the true perturbation set $\mathcal{U}(x)$ for a clean input x is a Gaussian around x', i.e. $\mathcal{N}(x', \sigma^2 I)$, for $\|x - x'\| \leq \gamma$. Given $\|x - x'\| \leq \gamma$, the total variation distance between the Gaussians around x and x' is bounded and is a function of γ and σ, i.e. $d(\gamma, \sigma)$[2]. Now by Theorems 2.4 and C.2, it suffices to do training on the perturbations coming from the representative distributions $\mathcal{R}(.)$ in order to get guarantees with respect to the true perturbation sets $\mathcal{U}(.)$.

Consequently, by Theorem C.2, for any class $\mathcal{H}$ and distribution $\mathcal{D}$, we get a training sample S_c of size $n = \mathcal{O}\Big(\frac{1}{\varepsilon^2}[VCdim(\mathcal{H}) \log(\frac{mk}{\varepsilon}) + \log(\frac{1}{\delta})]\Big)$, where for each $(x, y) \in S_c$, $m = \Omega(1/\varepsilon^2 \log(1/\varepsilon))$ perturbations are sampled from each of the distributions $u \in \mathcal{R}(x)$. Let S_p denote the set of all perturbations, then $S = S_c \cup S_p$. Given sample set S, with probability $\geq 1 - \delta$, for every $h \in \mathcal{H}$, $|\mathsf{DAL}_{\mathcal{D}}(h) - \mathsf{DAL}_S(h)| \leq d(\gamma, \sigma) + \varepsilon$. Where $\mathsf{DAL}_{\mathcal{D}}$ is defined with respect to the true perturbation sets $\mathcal{U}(.)$. This means, with probability at least $1 - \delta$,

$$\begin{aligned} &\mathsf{DAL}_{\mathcal{D}}(h) \\ &= \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \Big[\max_{x':\|x'-x\|\leq\gamma} \mathop{\mathbb{E}}_{\eta\sim\mathcal{N}(0,\sigma^2 I)} [h(x' + \eta) \neq y] \Big] \\ &\leq \mathsf{DAL}_S(h) + d(\gamma, \sigma) + \varepsilon \end{aligned}$$

which is equivalent to saying with high probability, the expected adversarial error over all the examples coming from $\mathcal{D}$ is bounded.

We can also use our framework to defend against multiple adversarial attacks. Consider an adversary that first alters the brightness of an image by scaling all its pixel values uniformly and then applies a bounded ℓ_2 perturbation. Suppose the brightness level varies between 0 and U. For each unperturbed input, we first adjust the brightness levels to $\{U/k, 2U/k, \ldots, U\}$ uniformly and then add Gaussian noise to each modified image. Now, by using k distributions for each unperturbed input in this manner, we can achieve robustness against multiple attacks.

2.5.3 Model II

In this section, we consider the scenario where the number of distribution perturbations is unbounded, however, there exists a set of *representative* distributions of size at most k such that each distribution $u \in \mathcal{U}(x)$, is completely covered by $\mathcal{R}(x)$, i.e. the probability density function of u is point-wise bounded by the maximum of probability density functions of the distributions in $\mathcal{R}(x)$. Here, the sampling procedure is similar to the one used in Section 2.5.1.Theorems 2.5 and C.3 exhibit the generalization guarantees in the realizable and agnostic settings.

Theorem 2.5 (realizable case)**.** *For any class $\mathcal{H}$ and distribution $\mathcal{D}$, a training sample S_c of size $n = \mathcal{O}\Big(\frac{1}{\varepsilon}[VCdim(\mathcal{H}) \log(\frac{mk}{\varepsilon}) + \log(\frac{1}{\delta}) + \frac{1}{\varepsilon}]\Big)$, where for each $(x, y) \in S_c$, $m = \Omega(1/\varepsilon^2 \log(1/\varepsilon))$ perturbations are sampled from each of the distributions $u \in \mathcal{R}(x)$. Let S_p denote the set of all perturbations, then $S = S_c \cup S_p$. Given sample set S, with probability $\geq 1-\delta$, every $h \in \mathcal{H}$ with $\mathsf{DAL}_S(h) = 0$ has $\mathsf{DAL}_{\mathcal{D}}(h) < k\varepsilon$.*

3 Derandomization

The classifiers discussed above add noise to the perturbed input to achieve robustness and hence are randomized. We modeled this phenomenon as the adversary getting to pick from a set of distributions over perturbed inputs, namely $\mathcal{U}(x)$, instead of a single perturbed input. Below we show how such classifiers can be derandomized in a general fashion while retaining the performance of the original classifier.

We derandomize the part where the classifiers add noise to the input. Thus, we model the adversary in the traditional sense, i.e., for the clean input x the adversary picks a perturbed input x' from an allowed set of perturbations $\mathcal{A}(x)$. Our classifier h takes the input x' and uses randomness R sampled according to $\mathcal{R}$ to make a prediction $h(x', R)$. Further, we also show how to derandomize a randomized certification procedure $\rho(x', R)$. The parameter $\rho(x', R)$ guarantees that all inputs within distance $\rho(x', R)$ from x', receive the same label as x'. As mentioned in the introduction, there are a few advantages to such a derandomization. First, in critical decisions, we desire that the output of our classifier be deterministic so that it always labels a given input with the same label (same for the certification radius). Second, such a derandomization could potentially be useful in boosting the robust accuracy of the classifier against various adversaries (more on this is discussed below).

The procedure is simple to describe: given the randomized classifier h which on input x' makes the prediction $h(x', R)$ (where R is the randomness sampled according to some distribution $\mathcal{R}$), we pre-sample multiple copies of the randomness needed during inference, say $R_1, R_2, \ldots, R_t$ and define a new *deterministic* classifier $h^{(R_1,R_2,\ldots,R_t)}$ as $\textsc{Maj}(h(x', R_i) \mid i \in [t])$. We show that when $t = \Omega(\log |\mathcal{A}|)$ then with high probability over

[2]Devroye et al. [2018](Proposition 2.2.) derive an upper bound on the total variation distance of two multivariate Gaussians.

the choice of $R_1, R_2, \ldots, R_t$ the deterministic classifier $h^{(R_1,R_2,\ldots,R_t)}$ has an adversarial loss related closely to the performance of the original classifier, i.e., $\mathsf{DAL}_{\mathcal{D}}(h)$. Recall that for the purposes of derandmization we are modeling the adversary in the traditional sense, i.e., for each clean input x the adversary is allowed to choose a perturbed input x' from an allowed set $\mathcal{A}(x)$. Hence, for a classifier h the definition of $\mathsf{DAL}_{\mathcal{D}}(h)$ is modified accordingly as

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[\max_{x'\in\mathcal{A}(x)}\left[\mathop{\mathbb{E}}_{R\sim\mathcal{R}}\mathbb{1}\left[h(x',R)\neq y\right]\right]\right].$$

Theorem 3.1 (Derandomizing a randomized robust classifier)**.** *Suppose h is a randomized classifier which on input x uses randomness $R\sim\mathcal{R}$ and outputs a label in $\mathcal{Y}$. Let $\mathcal{A}(x)$ be the set of perturbed versions of x from which the adversary chooses and define $\varepsilon(x,y) \coloneqq \max_{x'\in\mathcal{A}(x)}\left[\mathop{\mathbb{E}}_{R\sim\mathcal{R}}\mathbb{1}\left[h(x',R)\neq y\right]\right]$. Also, suppose that $\mathsf{DAL}_{\mathcal{D}}(h)\leq\varepsilon$, i.e., $\mathbb{E}_{(x,y)\sim\mathcal{D}}[\varepsilon(x,y)]\leq\varepsilon$, for some $\varepsilon>0$. For any $0<\eta<1/2$, let $\varepsilon(\eta)\coloneqq\Pr_{(x,y)\sim\mathcal{D}}[\varepsilon(x,y)\geq 1/2-\eta]$ (notice that $\varepsilon(\eta)\leq\frac{2\varepsilon}{1-2\eta}$). Let $\delta>0$ be a parameter. Set $t=\frac{100}{\eta^2}\cdot\log(|\mathcal{A}(x)|/\delta)$ and define $h^{(R_1,R_2,\ldots,R_t)}(x')\coloneqq\textsc{Maj}(h(x',R_i)\mid i\in[t])$. Then, with probability at least $1-\delta$ over the choice of $R_1,R_2,\ldots R_t$ sampled iid according to $\mathcal{R}$ we have*

$$\mathsf{DAL}_{\mathcal{D}}(h^{(R_1,R_2,\ldots,R_t)})\leq\delta+\varepsilon(\eta),$$

where

$$\begin{aligned}&\mathsf{DAL}_{\mathcal{D}}(h^{(R_1,R_2,\ldots,R_t)})\\&=\Pr_{(x,y)\sim\mathcal{D}}[\exists x'\in\mathcal{A}(x)\colon h^{(R_1,R_2,\ldots,R_t)}(x')\neq y].\end{aligned}$$

Boosting accuracy via Derandomization: Given a randomized classifier h, the procedure detailed in Theorem 3.1, fixes the randomness used during inference beforehand. However, during training, we do not modify h according to this pre-fixed randomness. It is natural to wonder if this knowledge of the pre-fixed randomness can be used during training to help boost the accuracy. We show in Section 4 that this is indeed the case empirically, by training our classifiers according to the pre-fixed randomness to be used during inference. An interesting question that arises here is whether under suitable assumptions a variant of Theorem 3.1 can be proven to capture the above phenomenon. Currently, Theorem 3.1 prescribes taking a majority vote over multiple instantiations of the original classifier h, whereas we would like to be able to train a classifier based on the pre-fixed randomness which might result in a classifier not corresponding to any instantiations of h.

Next we show that it is also possible to derandomize a randomized certification procedure. Suppose we have a classifier h and a randomized certification mechanism ρ which on input x' uses randomness R (sampled according to $\mathcal{R}$) and outputs a certifiable radius $\rho(x',R)$. We define $\rho^{(R_1,R_2,\ldots,R_t)}$ as the median of $\rho(R_i)$'s, i.e., $\textsc{Median}(\rho(R_i)\mid i\in[t])$ and this serves as our derandomized proxy for the certifiable radius. We show that the behavior of the median is statistically close to the behavior of $\rho(R)$. By this we mean the following: for any input x' let $\textsc{Robust}(h,x')$ denote a parameter which serves as the measure of the region around x' where the label according to the classifier h doesn't change. For instance, this could be the radius of the ℓ_p ball around x' in which the label according to h doesn't change. Our certification procedure ρ serves as an approximation to $\textsc{Robust}(h,x')$ and it is usually desired that with high probability $\rho(x',R)\in[(1-\beta)\,\textsc{Robust}(h,x'),(1+\alpha)\,\textsc{Robust}(h,x')]$ for some values of $\beta>0$ and $\alpha>0$ as close to 0 as possible. We are interested in preserving the property that $\rho(x',R)\in[(1-\beta)\,\textsc{Robust}(h,x'),(1+\alpha)\,\textsc{Robust}(h,x')]$.

Theorem 3.2 (Derandomizing a certifiably robust classifier)**.** *Let h be a hypothesis and ρ be a certification procedure which on input x' uses randomness $R\sim\mathcal{R}$ and outputs $\rho(x',R)\in\mathbb{R}_{\geq 0}$. Further, let $\textsc{Robust}(h,x')$ be as defined above and let $\gamma(\rho,x,y)$ be defined as*

$$\max_{x'\in\mathcal{A}(x)}\left[\mathop{\mathbb{E}}_{R\sim\mathcal{R}}\mathbb{1}\left[\frac{\rho(x',R)}{\textsc{Robust}(h,x')}\notin[1-\beta,1+\alpha]\right]\right]$$

for some positive parameters α, and β. For any $0<\eta<1/2$ define $\varepsilon(\eta)\coloneqq\Pr_{(x,y)\sim\mathcal{D}}[\gamma(\rho,x,y)\geq 1/2-\eta]$. Set $t=\frac{100}{\eta^2}\cdot\log(|\mathcal{A}(x)|/\delta)$ and define $\rho^{(R_1,R_2,\ldots,R_t)}(x')\coloneqq\textsc{Median}(h(x',R_i)\mid i\in[t])$. Let $\gamma(\rho^{(R_1,R_2,\ldots,R_t)},x,y)$ be defined as

$$\max_{x'\in\mathcal{A}(x)}\left[\mathbb{1}\left[\frac{\rho^{(R_1,R_2,\ldots,R_t)}(x')}{\textsc{Robust}(h,x')}\notin[1-\beta,1+\alpha]\right]\right].$$

Then, with probability $1-\delta$ over the choice of $R_1,R_2,\ldots R_t$ sampled iid according to $\mathcal{R}$ we have

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[\gamma(\rho^{(R_1,R_2,\ldots,R_t)},x,y)]\leq\varepsilon(\eta)+\delta.$$

4 Experiments

To show the effectiveness of our derandomization method in practice, we implement our framework on the method proposed by Dong and Xu [2023]. They use random projection filters to improve the adversarial robustness. In their method, they replace some of the CNN layers with random filters which are non-trainable, both during training and inference, and by doing so take away some of the adversarial impact on the input. More precisely, they re-instantiate the random filters for each training example and training is done only on the remaining weight

parameters of the CNN. Further, during inference time fresh random filters are used. In contrast, in our method, we fix the random filters to be used (in the style of Theorem 3.1) and train a few models with the pre-fixed convolutional filters. This fixing is done randomly at the beginning of the training. During the inference time, we use the majority vote of these trained models on the test input. We follow closely the experimental setup of Dong and Xu [2023] and refer the reader to their paper for further details . Below we outline the setup and our modification to implement the derandomization approach. Codes are publicly available at https://github.com/TTIC-Adversarial-Robustness/multiple_rpf.

4.1 Experimental Setup

Datases, Models and Computational Resources We use the CIFAR-10 dataset for our experiments that has 10 categories and contains $60K$ colored images with size 32×32, including 50K training images and 10K validation. We use the ResNet-18 architecture model on CIFAR-10. The random filters are applied at the first layer where the weights of the random filters are sampled as iid $N(0, 1/r^2)$ (r is the Kernel size). The ratio of the random filters to the total number of filters is kept at 0.75 (this parameter is denoted as N_r/N in Dong and Xu [2023]). For our experiments, we utilized a computing node with 24 cores, 192 GB of system RAM, and 4 NVIDIA GeForce RTX 2080 Ti GPUs, each with 11 GB of dedicated VRAM.

Training Strategy As we have described earlier, we have multiple models corresponding to each instantiation of the random coins tossed in the beginning. For training a *single* model, as in Dong and Xu [2023] we adhere to the established protocol of a state-of-the-art adversarial training strategy for setting up our experiments on CIFAR-10. The network undergoes training for 200 epochs with a batch size of 128 using SGD with a momentum of 0.9. The learning rate is fixed at 0.1, and the weight decay is set to 5×10^{-4}. Employing a piecewise decay learning rate scheduler, we initiate a decay factor of 0.1 at the 100^{th} and 150^{th} epochs. For generating adversarial examples, we utilize PGD-10 with a maximum perturbation size $\varepsilon = \frac{8}{255}$, and the step size of PGD is specified as $\frac{2}{255}$. Although, the training phase only considers PGD adversarial examples, yet during evaluation we consider other attacks.

Attacks To assess the adversarial robustness robustness of our algorithm, we utilize Projected Gradient Descent (PGD) Madry et al. [2018], Fast Gradient Sign Method (FGSM) Szegedy et al. [2013] and Auto Attack Croce and Hein [2020]. Following the standard protocol for attack configuration, we set the maximum perturbation size ε to $8/255$ for PGD, FGSM and Auto Attack. For PGD, the step size is established at $2/255$ over 20 steps.

Benchmark We compare various versions of our derandomized majority predictor model where the majority votes are taken over n base models each initialized with randomly chosen non-trainiable filters, with $n = 1, 2, \ldots, 14$. We also include a comparison with a variant of the model of Dong and Xu [2023] where a majority vote is taken over multiple inferences performed (once, 10, 20 and 30 times) on a single test input using fresh randomness each time (such a model with a majority over 5 votes was considered in Dong and Xu [2023]). This helps us better contrast the accuracy gains due to derandomization.

Results We can consistently match and outperform the benchmark by some percentage points in accuracy, even when the majority vote is taken over 30 trials for the benchmark, by just using 11 models with pre-fixed random filters. We tabulate the results in Table 1 below. Further plots detailing the results for varying number of trials can be found in the appendix.

Metric	Our Model	Dong and Xu [2023]
Natural Acc	0.8603	0.8580
PGD Acc	0.6302	0.6249
FGSM Acc	0.6600	0.6452
Auto Acc	0.6696	0.6591

Table 1: Comparison of Model Performances. The middle column corresponds to our model where we take the majority vote over 11 pre-fixed random filters. Second column corresponds to the model used by Dong and Xu [2023] where they take the majority vote over 30 fresh random filters.

Conclusion We studied the new notion of distributional adversarial loss and proved generalization guarantees for it, and showed how it derives sample complexity bounds for randomized smoothing methods. Furthermore, we show a general derandomization technique which preserves the extent of a randomized classifier's robustness and certification. In terms of further directions, as highlighted in Section 3, it is intriguing to explore the impact of employing pre-fixed randomness in the derandomization process, during training. Does this contribute to enhanced robustness? Our experiments, which incorporate random projection filters, provide some supporting evidence for this notion.

Acknowledgements

This work was supported in part by the National Science Foundation under grants CCF-2212968 and ECCS-2216899, by the Simons Foundation under the Simons Collaboration on the Theory of Algorithmic Fairness, and by the Defense Advanced Research Projects Agency under cooperative agreement HR00112020003, and by the Office of Naval Research MURI Grant N000142412742. The views expressed in this work do not necessarily reflect the position or the policy of the Government and no official endorsement should be inferred. Most of the research was done when Chen Dan was a postdoctoral researcher at TTIC and Prabhav Jain was an intern there. We thank Anand Bhattad for helpful discussions.

References

A. Athalye and N. Carlini. On the robustness of the cvpr 2018 white-box adversarial example defenses. *arXiv preprint arXiv:1804.03286*, 2018.

I. Attias, A. Kontorovich, and Y. Mansour. Improved generalization bounds for adversarially robust learning. *Journal of Machine Learning Research*, 23(175):1–31, 2022.

B. Biggio, I. Corona, D. Maiorca, B. Nelson, N. Šrndić, P. Laskov, G. Giacinto, and F. Roli. Evasion attacks against machine learning at test time. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2013, Prague, Czech Republic, September 23-27, 2013, Proceedings, Part III 13*, pages 387–402. Springer, 2013.

A. Blumer, D. Haussler, and M. K. Warmuth. Learnability and the vapnik-chervonenkis dimension. *Journal of the ACM (JACM)*, 36(4):929–965, 1989.

Y. Cao, N. Wang, C. Xiao, D. Yang, J. Fang, R. Yang, Q. A. Chen, M. Liu, and B. Li. Invisible for both camera and lidar: Security of multi-sensor fusion based perception in autonomous driving under physical-world attacks. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 176–194. IEEE, 2021.

N. Carlini and D. Wagner. Adversarial examples are not easily detected: Bypassing ten detection methods. In *Proceedings of the 10th ACM workshop on artificial intelligence and security*, pages 3–14, 2017.

J. Cohen, E. Rosenfeld, and Z. Kolter. Certified adversarial robustness via randomized smoothing. In *international conference on machine learning*, pages 1310–1320. PMLR, 2019.

F. Croce and M. Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 2206–2216. PMLR, 2020. URL `http://proceedings.mlr.press/v119/croce20b.html`.

D. Cullina, A. N. Bhagoji, and P. Mittal. Pac-learning in the presence of adversaries. *Advances in Neural Information Processing Systems*, 31, 2018.

L. Devroye, A. Mehrabian, and T. Reddad. The total variation distance between high-dimensional gaussians with the same mean. *arXiv preprint arXiv:1810.08693*, 2018.

M. Dong and C. Xu. Adversarial robustness via random projection filters. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4077–4086, 2023.

S. G. Finlayson, J. D. Bowers, J. Ito, J. L. Zittrain, A. L. Beam, and I. S. Kohane. Adversarial attacks on medical machine learning. *Science*, 363(6433):1287–1289, 2019.

I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. *International Conference on Learning Representations*, 2015.

D. Haussler, N. Littlestone, and M. K. Warmuth. Predicting $\{0, 1\}$-functions on randomly drawn points. *Information and Computation*, 115(2):248–292, 1994.

A. J. Levine and S. Feizi. Improved, deterministic smoothing for l_1 certified robustness. In *International Conference on Machine Learning*, pages 6254–6264. PMLR, 2021.

A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu. Towards deep learning models resistant to adversarial attacks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL `https://openreview.net/forum?id=rJzIBfZAb`.

J. Mohapatra, C.-Y. Ko, T.-W. Weng, P.-Y. Chen, S. Liu, and L. Daniel. Higher-order certification for randomized smoothing. *Advances in Neural Information Processing Systems*, 33:4501–4511, 2020.

O. Montasser, S. Hanneke, and N. Srebro. Vc classes are adversarially robustly learnable, but only improperly. In *Conference on Learning Theory*, pages 2512–2530. PMLR, 2019.

O. Montasser, S. Hanneke, and N. Srebro. Adversarially robust learning: A generic minimax optimal learner and characterization. *Advances in Neural Information Processing Systems*, 35:37458–37470, 2022.

H. Salman, J. Li, I. Razenshteyn, P. Zhang, H. Zhang, S. Bubeck, and G. Yang. Provably robust deep learning via adversarially trained smoothed classifiers. *Advances in Neural Information Processing Systems*, 32, 2019.

C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

V. Vapnik and A. Y. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and its Applications*, 16 (2):264, 1971.

R. Zhai, C. Dan, D. He, H. Zhang, B. Gong, P. Ravikumar, C.-J. Hsieh, and L. Wang. Macer: Attack-free and scalable robust training via maximizing certified radius. *International Conference on Learning Representations*, 2020.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes, Section 2]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes, we provide sample complexity results for our objective in the PAC-learning setting.]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]
2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   (b) Complete proofs of all theoretical results. [Yes, Appendix D]
   (c) Clear explanations of any assumptions. [Yes]
3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes, in the experiments section]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes, in Table 1, we report accuracies that are majority-vote of multiple runs of a model.]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes, Section 4.1, paragraph 1.]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes, Section 4]
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Yes, supplemental material]
   (d) Information about consent from data providers/curators. [Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]
5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

A Potential Broader Impact

Studying robustness to adversarial attacks is crucial since there is a threat that adversarial attacks are deployed in the real world. Making machine learning technologies resistant to adversarial attacks makes machine learning models more reliable to be deployed in areas such as automated driving and healthcare. The methods presented in this paper further our understanding of various mechanisms which enhance the robustness of classifiers.

B Further Details of Experimental Results

In Figure 1 the blue curve corresponds to the benchmark and the red curve corresponds to our model. The Figures 1(a) to 1(d) denote the performance for various adversaries, namely, no adversary (natural accuracy), FGSM, PGD and Auto Attack. We also collate the results in a bar graph, Figure 1(e).

C Missing Proofs

C.1 Proof of Lemma 2.1

Proof. For each input (x, y), distribution $u \in \mathcal{U}(x)$ and perturbation $z \sim u \in \mathcal{U}(x)$, define the random variable $Z_{(x,y,u,z)} = \mathbb{1}[h(z) \neq y]$. Let $Z_{(x,y,u)} = \sum_{z \in u \cap S'_{perturb}} \mathbb{1}[h(z) \neq y]$. Let $\hat{Z}_{(x,y,u)} = \mathbb{E}_{z \sim u} \mathbb{1}[h(z) \neq y]$. By Hoeffding bound, for each example (x, y) and $u \in \mathcal{U}(x)$,

$$\Pr\left[\left|\frac{1}{m} Z_{(x,y,u)} - \hat{Z}_{(x,y,u)}\right| \geq \varepsilon/8\right] = \Pr\left[\left|Z_{(x,y,u)} - m \cdot \hat{Z}_{(x,y,u)}\right| \geq \frac{m\varepsilon}{8}\right] \leq 2e^{-2(m\varepsilon/8)^2/m} = 2e^{-\frac{m\varepsilon^2}{32}}$$

Therefore, for a fixed example (x, y),

$$\Pr\left[\max_{u \in \mathcal{U}(x)} \left|\frac{1}{m} Z_{(x,y,u)} - \hat{Z}_{(x,y,u)}\right| \geq \varepsilon/8\right] \leq 2e^{-\frac{m\varepsilon^2}{32}} \tag{2}$$

which implies that:

$$\Pr\left[\left|\max_{u \in \mathcal{U}(x)} \frac{1}{m} Z_{(x,y,u)} - \max_{u \in \mathcal{U}(x)} \hat{Z}_{(x,y,u)}\right| \geq \varepsilon/8\right] \leq 2e^{-\frac{m\varepsilon^2}{32}}$$

Therefore, for each input (x, y), given $m \geq 32/\varepsilon^2 \log(200/\varepsilon)$:

$$\Pr_{r \sim \mathcal{R}}\left[\left|\max_{u \in \mathcal{U}(x)} \frac{1}{m} \sum_{z \in u \cap S'_{perturb}} \mathbb{1}[h(z) \neq y] - \max_{u \in \mathcal{U}(x)} \mathop{\mathbb{E}}_{z \sim u} \mathbb{1}[h(z) \neq y]\right| \geq \varepsilon/8\right] \leq \varepsilon/100 \tag{3}$$

where $r \sim \mathcal{R}$ is the randomness used for generating mk perturbations for each $(x, y) \in \text{supp}(\mathcal{D})$.

Let $Y(x, y)$ be an indicator random variable corresponding to input (x, y) defined as follows:

$$Y(x, y) = \mathbb{1}\left[\left|\max_{u \in \mathcal{U}(x)} \frac{1}{m} \sum_{z \in u \cap S'_{perturb}} \mathbb{1}[h(z) \neq y] - \max_{u \in \mathcal{U}(x)} \mathop{\mathbb{E}}_{z \sim u} \mathbb{1}[h(z) \neq y]\right| > \varepsilon/8\right]$$

Then:

$$\begin{aligned}
&\mathop{\mathbb{E}}_{(x,y) \sim \mathcal{D}} \mathop{\mathbb{E}}_{r \sim \mathcal{R}} \left|\max_{u \in \mathcal{U}(x)} \frac{1}{m} \sum_{z \in u \cap S'_{perturb}} \mathbb{1}[h(z) \neq y] - \max_{u \in \mathcal{U}(x)} \mathop{\mathbb{E}}_{z \sim u} \mathbb{1}[h(z) \neq y]\right| \\
&\leq \mathop{\mathbb{E}}_{(x,y) \sim \mathcal{D}} \Big[\Pr[Y(x, y) = 1](1) + \Pr[Y(x, y) = 0](\varepsilon/8)\Big] = \varepsilon/100 + \varepsilon/8 = 27\varepsilon/200
\end{aligned}$$

Therefore, by Markov's inequality:

$$\Pr_{r \sim \mathcal{R}}\left[\left|\mathop{\mathbb{E}}_{(x,y) \sim \mathcal{D}}\left[\max_{u \in \mathcal{U}(x)} \frac{1}{m} \sum_{z \in u \cap S'_{perturb}} \mathbb{1}[h(z) \neq y]\right] - \mathop{\mathbb{E}}_{(x,y) \sim \mathcal{D}}\left[\max_{u \in \mathcal{U}(x)} \mathop{\mathbb{E}}_{z \sim u} \mathbb{1}[h(z) \neq y]\right]\right| \geq \varepsilon/4\right] \leq 54/100 \tag{4}$$

Now, for each input (x, y), define the random variable $f(x, y) = \max_{u \in \mathcal{U}(x)} \frac{1}{m} \sum_{z \in u \cap S'_{perturb}} \mathbb{1}[h(z) \neq y]$. By Hoeffding bound:

$$\Pr_{S'_{clean} \sim \mathcal{D}^n}\left[\left|\frac{1}{n} \sum_{(x,y) \in S'_{clean}} f(x, y) - \mathop{\mathbb{E}}_{(x,y) \sim D} f(x, y)\right| \geq \varepsilon/4\right] \leq 2e^{-\frac{n\varepsilon^2}{8}} \tag{5}$$

Therefore, given $n \geq 13/\varepsilon^2 \geq \frac{8}{\varepsilon^2}\log(200/6)$:

$$\Pr_{S'_{clean}\sim\mathcal{D}^n}\left[\left|\frac{1}{n}\sum_{(x,y)\in S'_{clean}}\max_{u\in\mathcal{U}(x)}\frac{1}{m}\sum_{z\in u\cap S'_{perturb}}\mathbb{1}[h(z)\neq y]-\mathop{\mathbb{E}}_{(x,y)\sim D}\left[\max_{u\in\mathcal{U}(x)}\frac{1}{m}\sum_{z\in u\cap S'_{perturb}}\mathbb{1}[h(z)\neq y]\right]\right|\geq\varepsilon/4\right]$$
$$\leq 6/100 \tag{6}$$

Now, we define bad events

$$\mathcal{B}_1:\left|\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[\max_{u\in\mathcal{U}(x)}\frac{1}{m}\sum_{z\in u\cap S'_{perturb}}\mathbb{1}[h(z)\neq y]\right]-\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[\max_{u\in\mathcal{U}(x)}\mathop{\mathbb{E}}_{z\sim u}\mathbb{1}[h(z)\neq y]\right]\right|\geq\varepsilon/4$$

$$\mathcal{B}_2:\left|\frac{1}{n}\sum_{(x,y)\in S'_{clean}}\max_{u\in\mathcal{U}(x)}\frac{1}{m}\sum_{z\in u\cap S'_{perturb}}\mathbb{1}[h(z)\neq y]-\mathop{\mathbb{E}}_{(x,y)\sim D}\left[\max_{u\in\mathcal{U}(x)}\frac{1}{m}\sum_{z\in u\cap S'_{perturb}}\mathbb{1}[h(z)\neq y]\right]\right|\geq\varepsilon/4$$

Putting Equations (4) and (6) together,

$$\begin{aligned}
&\Pr_{S'_{clean}\sim\mathcal{D}^n,r\sim\mathcal{R}}\left[\left|\frac{1}{n}\sum_{(x,y)\in S'_{clean}}\max_{u\in\mathcal{U}(x)}\frac{1}{m}\sum_{z\in u\cap S'_{perturb}}\mathbb{1}[h(z)\neq y]-\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[\max_{u\in\mathcal{U}(x)}\mathop{\mathbb{E}}_{z\sim u}\mathbb{1}[h(z)\neq y]\right]\right|\geq\varepsilon/2\right]\\
\leq&\Pr_{S'_{clean}\sim\mathcal{D}^n,r\sim\mathcal{R}}[\mathcal{B}_1=1]+\Pr_{S'_{clean}\sim\mathcal{D}^n,r\sim\mathcal{R}}[\mathcal{B}_2=1]\\
\leq&54/100+\Pr_{r\sim\mathcal{R}}[r]\cdot\Pr_{S'_{clean}\sim\mathcal{D}^n}[B_2=1|r]\\
\leq&54/100+6/100=3/5
\end{aligned}\tag{7}$$

where Equation (7) holds for all $h\in\mathcal{H}$. Now, $\Pr(B)\geq\Pr(A,B)=\Pr(A)\Pr(B|A)$. Consider drawing set S and suppose event A occurs. Let h be in $\mathcal{H}$ such that:

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[\max_{u\in\mathcal{U}(x)}\left[\mathop{\mathbb{E}}_{z\sim u}\mathbb{1}\left[h(z)\neq y\right]\right]\right]\geq\varepsilon,\frac{1}{n}\sum_{(x,y)\in S_{clean}}\left[\max_{u\in\mathcal{U}(x)}\left[\frac{1}{m}\sum_{z\in\mathcal{U}(x)\cap S_{perturb}}\mathbb{1}\left[h(z)\neq y\right]\right]\right]=0$$

By Equation (7), given $\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\max_{u\in\mathcal{U}(x)}\left[\mathbb{E}_{z\sim u}\mathbb{1}\left[h(z)\neq y\right]\right]\right]\geq\varepsilon$, when $m=\Omega(1/\varepsilon^2\cdot\log(1/\varepsilon)), n=\Omega(1/\varepsilon^2)$,

$$\Pr\left[\frac{1}{n}\sum_{(x,y)\in S'_{clean}}\max_{u\in\mathcal{U}(x)}\frac{1}{m}\sum_{z\in u\cap S'_{perturb}}\mathbb{1}[h(z)\neq y]\leq\varepsilon/2\right]\leq 3/5$$

.

Thus, $\Pr(B|A)\geq 2/5$ and $\Pr(B)\geq(2/5)\Pr(A)$ as desired. □

C.2 Proof of Theorem 2.2

Proof. Consider drawing a set S_c of n examples from $\mathcal{D}$ and then for each $(x,y)\in S_c$, add m perturbations sampled from each $u\in\mathcal{U}(x)$ to S_p and let $S=S_c\cup S_p$. Define A as the event where there exists $h\in\mathcal{H}$ with $\mathsf{DAL}_{\mathcal{D}}(h)>\varepsilon$ but $\mathsf{DAL}_S(h)=0$. By Lemma 2.1, it suffices to prove that $\Pr(B)\leq(2/5)\delta$, where B is the same event as defined in Lemma 2.1. Consider a third experiment. Draw a set S''_c of $2n$ points from $\mathcal{D}$ and then augment each natural example (x,y) by adding mk perturbations sampled from each $u\in\mathcal{U}(x)$ to get a set S''. Now, in fact, the set $S''=\{B_1,\cdots,B_{2n}\}$ where each $B_i\in S''$ is a ball around the i^{th} clean example in S''_c that contains the i^{th} clean example and all its $m\cdot k$ perturbations. Next, randomly partition S'' into two sets S and S' of n balls each.

Let B^* denote the event that there exists $h \in \mathcal{H}$ with $\mathsf{DAL}_S(h) = 0$ but $\mathsf{DAL}_{S'}(h) \geq \varepsilon/2$. $\Pr(B^*) = \Pr(B)$ since drawing $2n$ points from $\mathcal{D}$ then augmenting them by adding $m \cdot k$ perturbations per each natural example, and randomly partitioning them into two sets of size n, results in the same distribution on (S, S') as does drawing S and S' directly. The advantage of this new experiment is that we can now argue that $\Pr(B^*)$ is low, with probability now taken over just the random partition of S'' into S and S'. The key point is that since S'' is fixed, there are at most $|\mathcal{H}[S'']| \leq \mathcal{H}[2n \cdot m \cdot k]$ events to worry about. Specifically, it suffices to prove that for any fixed $h \in \mathcal{H}[S'']$, the probability over the partition of S'' that h makes zero distributional adversarial loss on S but $\mathsf{DAL}_{S'}(h) \geq \varepsilon/2$ is at most $2\delta/(5\mathcal{H}[2n \cdot m \cdot k])$.We can then apply the union bound.

Consider the following specific method for partitioning S'' into S and S'. Randomly put the balls in S'' into pairs: $(a_1, b_1), (a_2, b_2), \cdots, (a_n, b_n)$. For each index i, flip a fair coin. If heads put a_i in S and b_i into S', else if tails put a_i into S' and b_i into S. Now, fix some partition $h \in \mathcal{H}[S'']$ and consider the probability over these n fair coin flips such $\mathsf{DAL}_S(h) = 0$ that but $\mathsf{DAL}_{S'}(h) \geq \varepsilon/2$. First of all, if for any index i, h makes a robustness mistake, i.e. a mistake on any examples inside a ball, on both a_i and b_i then the probability is zero (because it cannot have zero robust loss on S). Second, if there are fewer than $\varepsilon n/2$ indices i such that h makes a robustness mistake on either a_i or b_i then again the probability is zero because it cannot possibly be the case that $\mathsf{DAL}_{S'}(h) \geq \varepsilon/2$. So, assume there are $r \geq \varepsilon n/2$ indices i such that h makes a robustness mistake on exactly one of a_i or b_i. In this case, the chance that all of those mistakes land in S' is exactly $1/2^r$. This quantity is at most $1/2^{\varepsilon n/2} \leq 2\delta/(5\mathcal{H}[2n \cdot m \cdot k])$ as desired for n given in the theorem statement. □

C.3 Proof of Theorem 2.1

Proof. We use Sauer's Lemma and Theorem 2.2 to complete the proof.

Using Sauer's lemma, we have that for $\mathcal{H}[2n \cdot m \cdot k] \leq (2enmk/d)^d$. From Theorem 2.2 we have the following:

$$n \geq \frac{2}{\varepsilon}[\log(2\mathcal{H}[2n \cdot m \cdot k]) + \log(\frac{1}{\delta}) + \frac{7}{\varepsilon}]$$

Combining with Sauer's lemma implies that:

$$n \geq \frac{2}{\varepsilon}[\log(2(\frac{2enmk}{d})^d) + \log(\frac{1}{\delta}) + \frac{7}{\varepsilon}]$$

Therefore,

$$n \geq \frac{2}{\varepsilon}\Big[d\log(n) + d\log(2emk/d) + d\log(2) + \log(1/\delta) + \frac{7}{\varepsilon}\Big]$$

Therefore, it is sufficient to have:

$$n \geq \frac{2}{\varepsilon}\Big[d\log(n) + d\log(2emk/d) + d + \log(1/\delta) + \frac{7}{\varepsilon}\Big] \tag{8}$$

$$n \geq \frac{2d}{\varepsilon}\log(n) + \frac{2d}{\varepsilon}\log(2emk/d) + \frac{2d}{\varepsilon} + \frac{2}{\varepsilon}\log(1/\delta) + \frac{14}{\varepsilon^2} \tag{9}$$

In order to have $x \geq a\log(x) + b$ it is sufficient to have $x \geq 4a\log(2a) + 2b$. Therefore, in order to have Equation (13), it is sufficient to have:

$$n \geq \frac{8d}{\varepsilon}\log(\frac{8d}{\varepsilon}) + \frac{4d}{\varepsilon}\log(2emk/d) + \frac{4d}{\varepsilon} + \frac{4}{\varepsilon}\log(1/\delta) + \frac{28}{\varepsilon^2}$$

And it is sufficient to have:

$$n \geq \frac{8d}{\varepsilon}\log(\frac{16demk}{d\varepsilon}) + \frac{4d}{\varepsilon} + \frac{4}{\varepsilon}\log(1/\delta) + \frac{28}{\varepsilon^2}$$

Therefore:

$$n = \mathcal{O}\Big(\frac{1}{\varepsilon} \cdot d\log(\frac{mk}{\varepsilon}) + \frac{1}{\varepsilon}\log(\frac{1}{\delta}) + \frac{1}{\varepsilon^2}\Big)$$

□

C.4 Proof of Theorem 2.3

Putting together Lemma C.1 and Theorem C.1 proves Theorem 2.3 holds. Similar to Lemma 2.1, we prove the following lemma in the agnostic case.

Lemma C.1. *Let $\mathcal{H}$ be a concept class over a domain $\mathcal{X}$. Let S_c and S_c' be sets of n elements drawn from some distribution $\mathcal{D}$ over $\mathcal{X}$, where $n \geq 13/\varepsilon^2$. For each $(x, y) \in S_c$, $m = \Omega(1/\varepsilon^2 \log(1/\varepsilon))$ perturbations sampled from each $u \in \mathcal{U}(x)$ are added to a set S_p and finally $S = S_c \cup S_p$. Similarly, S_c' is augmented to get S'. Let A denote the event that there exists $h \in \mathcal{H}$ such that $|\mathsf{DAL}_{\mathcal{D}}(h) - \mathsf{DAL}_S(h)| \geq \varepsilon$. Let B denote the event that there exists $h \in \mathcal{H}$ such that $|\mathsf{DAL}_{S'}(h) - \mathsf{DAL}_S(h)| \geq \varepsilon/2$. Then $\Pr(B) \geq (2/5)\Pr(A)$.*

Proof. We need to follow an approach similar to the proof of Lemma 2.1 with a minor modification in the last step. Similar to Equation (7), we have for all $h \in \mathcal{H}$,

$$\Pr_{\substack{S_c' \sim \mathcal{D}^n, \\ r \sim \mathcal{R}}} \left[\left| \frac{1}{n} \sum_{(x,y) \in S_c'} \max_{u \in \mathcal{U}(x)} \frac{1}{m} \sum_{z \in u \cap S_p'} \mathbb{1}[h(z) \neq y] - \mathop{\mathbb{E}}_{(x,y) \sim \mathcal{D}} \left[\max_{u \in \mathcal{U}(x)} \mathop{\mathbb{E}}_{z \sim u} \mathbb{1}[h(z) \neq y] \right] \right| \leq \varepsilon/2 \right] \geq 2/5 \tag{10}$$

which is same as saying:

$$\Pr \left[|\mathsf{DAL}_{\mathcal{D}}(h) - \mathsf{DAL}_{S'}(h)| \leq \varepsilon/2 \right] \geq 2/5 \tag{11}$$

Now, $\Pr(B) \geq \Pr(A, B) = \Pr(A)\Pr(B|A)$. Consider drawing set S and suppose event A occurs, let h be in $\mathcal{H}$ such that $|\mathsf{DAL}_{\mathcal{D}}(h) - \mathsf{DAL}_S(h)| \geq \varepsilon$. By triangle's inequality and Equation (11), $\Pr(B|A) = \Pr[|\mathsf{DAL}_{S'}(h) - \mathsf{DAL}_S(h)| \geq \varepsilon/2 | A] \geq 2/5$. Therefore, we can conclude that $\Pr(B) \geq (2/5)\Pr(A)$.

□

Theorem C.1. *For any class $\mathcal{H}$ and distribution $\mathcal{D}$, a training sample S_c of size*

$$n \geq \frac{13}{\varepsilon^2}[\log(5\mathcal{H}[2n \cdot m \cdot k]) + \log(\frac{1}{\delta})]$$

where for each $(x, y) \in S_c$, $m = \Omega(1/\varepsilon^2 \log(1/\varepsilon))$ perturbations are sampled from each of the distributions $u \in \mathcal{U}(x)$. Let S_p denote the set of all perturbations, then $S = S_c \cup S_p$. Let $k = \max_{x \in \mathcal{X}} \mathcal{U}(x)$. Then with probability at least $1 - \delta$, every $h \in \mathcal{H}$ will have $|\mathsf{DAL}_{\mathcal{D}}(h) - \mathsf{DAL}_S(h)| \leq \varepsilon$.

Proof of Theorem C.1. By Lemma C.1, it suffices to prove $\Pr(B) \leq (2/5)\delta$. Consider a third experiment. Draw a set S_c'' of $2n$ points from $\mathcal{D}$ and then augment each natural example (x, y) by adding $m \cdot k$ perturbations sampled from each $u \in \mathcal{U}(x)$ to get a set S''. Now, in fact, the set $S'' = \{B_1, \cdots, B_{2n}\}$ where each $B_i \in S''$ is a ball around the i^{th} clean example in S_c'' that contains the i^{th} clean example and all its $m \cdot k$ perturbations. Next, randomly partition S'' into two sets S and S' of n balls each.

Let B^* be the event that there exists $h \in \mathcal{H}[S'']$ such that $|\mathsf{DAL}_S(h) - \mathsf{DAL}_{S'}(h)| > \varepsilon/2$. Consider an experiment where we randomly put the balls in S'' into pairs (a_i, b_i). For each index i, flip a fair coin. If heads put a_i in S and b_i into S', else if tails put a_i into S' and b_i into S. Consider the the value of $\mathsf{DAL}_{S'} - \mathsf{DAL}_S$ and see how it changes as we flip coins for $i = 1, \cdots, n$. Initially, the difference is zero. For a fixed pair (a_i, b_i), suppose the difference between $\mathsf{DAL}_{a_i}(h) - \mathsf{DAL}_{b_i}(h) = \eta$ for some value of η between $[-1, 1]$. When the i^{th} random coin is flipped, the difference $\mathsf{DAL}_{S'}(h) - \mathsf{DAL}_S(h)$ increases by η with probability $1/2$ and decreases by η with probability $1/2$. The probability that when taking a random walk of n steps where each step has a length of at most 1, we end up more than $\varepsilon n/2$ steps away from the origin, is at the most the probability that among n coin flips the number of heads differs from its expectation by more than $\varepsilon n/4$. By Hoeffding bounds, this is at most $2e^{-\varepsilon^2 n/8}$. This quantity is at most $2\delta/5\mathcal{H}[2nmk]$ as desired for $n \geq (8/\varepsilon^2)(\log(5\mathcal{H}[2nmk]) + \log(1/\delta))$. By applying union bound, $\Pr(B) \leq (2/5)\delta$. By applying Lemma C.1, since $n \geq 13/\varepsilon^2$ and $m = \Omega(1/\varepsilon^2 \log(1/\varepsilon))$, $\Pr(A) \leq (5/2)\Pr(B)$ which implies that $\Pr(A) \leq \delta$. □

Proof of Theorem 2.3. We use Sauer's Lemma and Theorem C.1 to complete the proof.

Using Sauer's lemma, we have that for $\mathcal{H}[2n \cdot m \cdot k] \leq (2enmk/d)^d$. From Theorem C.1 we have the following:

$$n \geq \frac{13}{\varepsilon^2}[\log(2\mathcal{H}[2n \cdot m \cdot k]) + \log(\frac{1}{\delta})]$$

Combining with Sauer's lemma implies that:

$$n \geq \frac{13}{\varepsilon^2}[\log(2(\frac{2enmk}{d})^d) + \log(\frac{1}{\delta})]$$

Therefore,

$$n \geq \frac{13}{\varepsilon^2}\Big[d\log(n) + d\log(2emk/d) + d\log(2) + \log(1/\delta)\Big]$$

Therefore, it is sufficient to have:

$$n \geq \frac{13}{\varepsilon^2}\Big[d\log(n) + d\log(2emk/d) + d + \log(1/\delta)\Big] \tag{12}$$

$$n \geq \frac{13d}{\varepsilon^2}\log(n) + \frac{13d}{\varepsilon^2}\log(2emk/d) + \frac{13d}{\varepsilon^2} + \frac{13}{\varepsilon^2}\log(1/\delta) \tag{13}$$

In order to have $x \geq a\log(x) + b$ it is sufficient to have $x \geq 4a\log(2a) + 2b$. Therefore, in order to have Equation (13), it is sufficient to have:

$$n \geq \frac{52d}{\varepsilon^2}\log(\frac{26d}{\varepsilon^2}) + \frac{26d}{\varepsilon^2}\log(2emk/d) + \frac{26d}{\varepsilon^2} + \frac{26}{\varepsilon^2}\log(1/\delta)$$

And it is sufficient to have:

$$n \geq \frac{52d}{\varepsilon^2}\log(\frac{52demk}{d\varepsilon^2}) + \frac{26d}{\varepsilon^2} + \frac{26}{\varepsilon^2}\log(1/\delta)$$

Therefore:

$$n = \mathcal{O}\Big(\frac{1}{\varepsilon^2} \cdot d\log(\frac{mk}{\varepsilon}) + \frac{1}{\varepsilon^2}\log(\frac{1}{\delta})\Big)$$

□

C.5 Proof of Theorem 2.4 and extension to the agnostic case Theorem C.2

Proof of Theorem 2.4. In order to prove Theorem 2.4, they key idea is to prove Lemma C.2, which states the following: Let A denote the event that there exists $h \in \mathcal{H}$ with zero empirical distributional adversarial error on S (with respect to the representative perturbation sets $\mathcal{R}(.)$) but true distributional adversarial error at least ε (with respect to the true perturbation sets $\mathcal{U}(.)$). Now draw a *fresh* test set S' of n examples from $\mathcal{D}$ where for each example $(x, y) \in S'$, m perturbations are drawn from each distribution $u \in \mathcal{R}(x)$ and are added to S'. Let B denote the event that there exists $h \in \mathcal{H}$ with zero distributional adversarial loss on S but distributional adversarial loss at least $\varepsilon/2$ on S'. We prove that $\Pr(B) \geq (2/5)\Pr(A)$.

Lemma C.2. *Let $\mathcal{H}$ be a concept class over a domain $\mathcal{X}$. Let S_c and S_c' be sets of n elements drawn from some distribution $\mathcal{D}$ over $\mathcal{X}$, where $n = \Omega(1/\varepsilon^2)$. For each $(x, y) \in S_c$, $m = \Omega(1/\varepsilon^2 \log(1/\varepsilon))$ perturbations sampled from each $u \in \mathcal{R}(x)$ are added to a set S_p and finally $S = S_c \cup S_p$. Similarly, S_c' is augmented to get S'. Let A denote the event that there exists $h \in \mathcal{H}$ with zero empirical distributional adversarial error on S but true distributional adversarial error $\geq \varepsilon + \varepsilon'$:*

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[\max_{u\in\mathcal{U}(x)}\left[\mathop{\mathbb{E}}_{z\sim u}\mathbb{1}\left[h(z) \neq y\right]\right]\right] \geq \varepsilon + \varepsilon', \quad \frac{1}{n}\sum_{(x,y)\in S_c}\left[\max_{u\in\mathcal{R}(x)}\left[\frac{1}{m}\sum_{z\in\mathcal{R}(x)\cap S_p}\mathbb{1}\left[h(z) \neq y\right]\right]\right] = 0$$

Let B denote the event that there exists $h \in \mathcal{H}$ with zero distributional adversarial loss on S but distributional adversarial loss $\geq \varepsilon/2$ on S':

$$\frac{1}{n}\sum_{(x,y)\in S'_c}\left[\max_{u\in\mathcal{R}(x)}\left[\frac{1}{m}\sum_{z\in\mathcal{R}(x)\cap S'_p}\mathbb{1}\left[h(z)\neq y\right]\right]\right] \geq \varepsilon/2,$$

$$\frac{1}{n}\sum_{(x,y)\in S_c}\left[\max_{u\in\mathcal{R}(x)}\left[\frac{1}{m}\sum_{z\in\mathcal{R}(x)\cap S_p}\mathbb{1}\left[h(z)\neq y\right]\right]\right] = 0$$

Then $\Pr(B) \geq (2/5)\Pr(A)$.

Proof of Lemma C.2. Suppose event A happens, and let h be in $\mathcal{H}$ such that $\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\max_{u\in\mathcal{U}(x)}\left[\mathbb{E}_{z\sim u}\mathbb{1}\left[h(z)\neq y\right]\right]\right] \geq \varepsilon + \varepsilon'$. Let f be a function that maps an input (x, y) to a distribution $u \in \mathcal{U}(x)$ with maximum error given classifier h:

$$f : (x,y) \to u \in \mathcal{U}(x) : u = \operatorname*{argmax}_{\mathcal{U}(x)} \mathop{\mathbb{E}}_{z\sim u}\mathbb{1}[h(z)\neq y]$$

, and

$$f' : (x,y) \to r \in \mathcal{R}(x) : r = \operatorname*{argmin}_{\mathcal{R}(x)} TV(r, f(x,y))$$

Given event A, we know:

$$\varepsilon + \varepsilon' \leq \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[\mathop{\mathbb{E}}_{z\sim f(x,y)}\mathbb{1}\left[h(z)\neq y\right]\right] \leq \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[\mathop{\mathbb{E}}_{z\sim f'(x,y)}\mathbb{1}\left[h(z)\neq y\right]\right] + \varepsilon' \tag{14}$$

Therefore:

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[\mathop{\mathbb{E}}_{z\sim f'(x,y)}\mathbb{1}\left[h(z)\neq y\right]\right] \geq \varepsilon \tag{15}$$

which implies that:

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[\max_{u\in\mathcal{R}(x)}\left[\mathop{\mathbb{E}}_{z\sim u}\mathbb{1}\left[h(z)\neq y\right]\right]\right] \geq \varepsilon \tag{16}$$

Combining Equation (16) with Lemma 2.1 implies that event B happens with probability at least $(2/5)\Pr(A)$. □

The rest of the proof of Theorem 2.4 is similar to the proof of Theorem 2.1. First, we use the application of Lemma C.2 to argue about the distributional adversarial loss on the test data with perturbations generated from representative distributions $\mathcal{R}(.)$ instead of population distributional adversarial loss with respect to $\mathcal{U}(.)$. Furthermore, similar to the proof of Theorem 2.2, we can show that for large enough test and training data coming from the same distribution, it cannot be the case that there is a large gap between training and test distributional adversarial loss. In the end, the application of Sauer's lemma similar to the proof of Theorem 2.1 completes the proof. □

Theorem C.2 (agnostic case)**.** *For any class $\mathcal{H}$ and distribution $\mathcal{D}$, a training sample S_c of size $n = \mathcal{O}\Big(\frac{1}{\varepsilon^2}[VCdim(\mathcal{H})\log(\frac{mk}{\varepsilon}) + \log(\frac{1}{\delta})]\Big)$, where for each $(x, y) \in S_c$, $m = \Omega(1/\varepsilon^2\log(1/\varepsilon))$ perturbations are sampled from each of the distributions $u \in \mathcal{R}(x)$. Let S_p denote the set of all perturbations, then $S = S_c \cup S_p$. Given sample set S, with probability $\geq 1-\delta$, for every $h \in \mathcal{H}$, $|\mathsf{DAL}_{\mathcal{D}}(h) - \mathsf{DAL}_S(h)| \leq \varepsilon' + \varepsilon$.*

Proof of Theorem C.2. In the agnostic case, first similar to Lemma C.2, we can show the following lemma holds.

Lemma C.3. *Let $\mathcal{H}$ be a concept class over a domain $\mathcal{X}$. Let S_c and S'_c be sets of n elements drawn from some distribution $\mathcal{D}$ over $\mathcal{X}$, where $n \geq 13/\varepsilon^2$. For each $(x, y) \in S_c$, $m = \Omega(1/\varepsilon^2\log(1/\varepsilon))$ perturbations sampled from each $u \in \mathcal{R}(x)$ are added to a set S_p and finally $S = S_c \cup S_p$. Similarly, S'_c is augmented to get S'. Let A denote the event that there exists $h \in \mathcal{H}$ such that $|\mathsf{DAL}_{\mathcal{D}}(h) - \mathsf{DAL}_S(h)| \geq \varepsilon + \varepsilon'$. Let B denote the event that there exists $h \in \mathcal{H}$ such that $|\mathsf{DAL}_{S'}(h) - \mathsf{DAL}_S(h)| \geq \varepsilon/2$. Then $\Pr(B) \geq (2/5)\Pr(A)$.*

The rest of the proof of Theorem C.2 is similar to the proof of Theorem 2.3. First, we use the application of Lemma C.3 to argue about the distributional adversarial loss on the test data with perturbations generated from representative distributions $\mathcal{R}(.)$ instead of population distributional adversarial loss with respect to $\mathcal{U}(.)$. Furthermore, similar to the proof of Theorem C.1, we can show that for large enough test and training data coming from the same distribution, it cannot be the case that there is a large gap between training and test distributional adversarial loss. In the end, the application of Sauer's lemma similar to the proof of Theorem 2.3 completes the proof. □

C.6 Proof of Theorem 2.5 and extension to the agnostic case

Proof of Theorem 2.5. In order to prove Theorem 2.5 holds, first we prove the following lemma:

Lemma C.4. *Let $\mathcal{H}$ be a concept class over a domain $\mathcal{X}$. Let S_c and S_c' be sets of n elements drawn from some distribution $\mathcal{D}$ over $\mathcal{X}$, where $n = \Omega(1/\varepsilon^2)$.For each $(x, y) \in S_c$, $m = \Omega(1/\varepsilon^2 \log(1/\varepsilon))$ perturbations sampled from each $u \in \mathcal{R}(x)$ are added to a set S_p and finally $S = S_c \cup S_p$. Similarly, S_c' is augmented to get S'. Let A denote the event that there exists $h \in \mathcal{H}$ with zero empirical distributional adversarial error on S but true distributional adversarial error $\geq k\varepsilon$:*

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[\max_{u\in\mathcal{U}(x)}\left[\mathop{\mathbb{E}}_{z\sim u}\mathbb{1}\left[h(z)\neq y\right]\right]\right] \geq k\varepsilon,$$

$$\frac{1}{n}\sum_{(x,y)\in S_c}\left[\max_{u\in\mathcal{R}(x)}\left[\frac{1}{m}\sum_{z\in\mathcal{R}(x)\cap S_p}\mathbb{1}\left[h(z)\neq y\right]\right]\right] = 0$$

Let B denote the event that there exists $h \in \mathcal{H}$ with zero distributional adversarial loss on S but distributional adversarial loss $\geq \varepsilon/2$ on S':

$$\frac{1}{n}\sum_{(x,y)\in S_c'}\left[\max_{u\in\mathcal{R}(x)}\left[\frac{1}{m}\sum_{z\in\mathcal{R}(x)\cap S_p'}\mathbb{1}\left[h(z)\neq y\right]\right]\right] \geq \varepsilon/2,$$

$$\frac{1}{n}\sum_{(x,y)\in S_c}\left[\max_{u\in\mathcal{R}(x)}\left[\frac{1}{m}\sum_{z\in\mathcal{R}(x)\cap S_p}\mathbb{1}\left[h(z)\neq y\right]\right]\right] = 0$$

Then $\Pr(B) \geq (2/5)\Pr(A)$.

Proof of Lemma C.4. Suppose event A happens, and let h be in $\mathcal{H}$ such that $\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\max_{u\in\mathcal{U}(x)}\left[\mathbb{E}_{z\sim u}\mathbb{1}\left[h(z)\neq y\right]\right]\right] \geq k\varepsilon$. Let f be a function that maps an input (x, y) to a distribution $u \in \mathcal{U}(x)$ with maximum error given classifier h:

$$f : (x, y) \to u \in \mathcal{U}(x) : u = \mathop{\mathrm{argmax}}_{\mathcal{U}(x)} \mathop{\mathbb{E}}_{z\sim u} \mathbb{1}[h(z) \neq y],$$

and

$$f' : (x, y) \to r \in \mathcal{R}(x) : r \text{ has the maximum coverage of the error region of } f(x, y)$$

.

Given event A, we know:

$$\varepsilon \leq \frac{1}{k}\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[\mathop{\mathbb{E}}_{z\sim f(x,y)}\mathbb{1}\left[h(z)\neq y\right]\right] \leq \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[\mathop{\mathbb{E}}_{z\sim f'(x,y)}\mathbb{1}\left[h(z)\neq y\right]\right] \tag{17}$$

where the last inequality holds by the pigeon-hole principle. Therefore $\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\mathbb{E}_{z\sim f'(x,y)}\mathbb{1}\left[h(z)\neq y\right]\right] \geq \varepsilon$, which implies that:

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[\max_{u\in\mathcal{R}(x)}\left[\mathop{\mathbb{E}}_{z\sim u}\mathbb{1}\left[h(z)\neq y\right]\right]\right] \geq \varepsilon$$

Combining it with Lemma 2.1 implies that event B happens with probability at least $(2/5)\Pr(A)$. □

The rest of the proof of Theorem 2.5 is similar to Theorem 2.1. First, we use the application of Lemma C.4 to argue about the distributional adversarial loss on the test data with perturbations generated from representative distributions $\mathcal{R}(.)$ instead of population distributional adversarial loss with respect to $\mathcal{U}(.)$. Furthermore, similar to the proof of Theorem 2.2, we can show that for large enough test and training data coming from the same distribution, it cannot be the case that there is a large gap between training and test distributional adversarial loss. In the end, the application of Sauer's lemma similar to the proof of Theorem 2.1 completes the proof. □

Theorem C.3 (agnostic case)**.** *For any class $\mathcal{H}$ and distribution $\mathcal{D}$, a training sample S_c of size $n = \mathcal{O}\Big(\frac{1}{\varepsilon^2}[VCdim(\mathcal{H})\log(\frac{mk}{\varepsilon}) + \log(\frac{1}{\delta})]\Big)$, where for each $(x, y) \in S_c$, $m = \Omega(1/\varepsilon^2 \log(1/\varepsilon))$ perturbations are sampled from each of the distributions $u \in \mathcal{R}(x)$. Let S_p denote the set of all perturbations, then $S = S_c \cup S_p$. Given sample set S, with probability $\geq 1 - \delta$, for every $h \in \mathcal{H}$, $|\mathsf{DAL}_{\mathcal{D}}(h) - \mathsf{DAL}_S(h)| \leq k\varepsilon$.*

Proof of Theorem C.3. In the agnostic case, first similar to Lemma C.4 we can show the following lemma holds:

Lemma C.5. *Let $\mathcal{H}$ be a concept class over a domain $\mathcal{X}$. Let S_c and S_c' be sets of n elements drawn from some distribution $\mathcal{D}$ over $\mathcal{X}$, where $n \geq 13/\varepsilon^2$. For each $(x, y) \in S_c$, $m = \Omega(1/\varepsilon^2 \log(1/\varepsilon))$ perturbations sampled from each $u \in \mathcal{R}(x)$ are added to a set S_p and finally $S = S_c \cup S_p$. Similarly, S_c' is augmented to get S'. Let A denote the event that there exists $h \in \mathcal{H}$ such that $|\mathsf{DAL}_{\mathcal{D}}(h) - \mathsf{DAL}_S(h)| \geq k\varepsilon$. Let B denote the event that there exists $h \in \mathcal{H}$ such that $|\mathsf{DAL}_{S'}(h) - \mathsf{DAL}_S(h)| \geq \varepsilon/2$. Then $\Pr(B) \geq (2/5)\Pr(A)$.*

The rest of the proof of Theorem C.3 is similar to the proof Theorem 2.3. First, we use the application of Lemma C.5 to argue about the distributional adversarial loss on the test data with perturbations generated from representative distributions $\mathcal{R}(.)$ instead of population distributional adversarial loss with respect to $\mathcal{U}(.)$. Furthermore, similar to the proof of Theorem C.1, we can show that for large enough test and training data coming from the same distribution, it cannot be the case that there is a large gap between training and test distributional adversarial loss. In the end, the application of Sauer's lemma similar to the proof of Theorem 2.3 completes the proof. □

C.7 Proof of Theorem 3.1

Proof. Fix a sample (x, y) such that $\varepsilon(x, y) \leq 1/2 - \eta$. We will show that

$$\Pr_{R_1,R_2,\ldots,R_t}[\exists x' \in \mathcal{A}(x)\colon h^{(R_1,R_2,\ldots,R_t)}(x') \neq y] \leq \delta.$$

For this fix we fix an $x' \in \mathcal{A}(x)$ and analyze

$$\Pr_{R_1,R_2,\ldots,R_t}[h^{(R_1,R_2,\ldots,R_t)}(x') \neq y].$$

Notice that $\Pr_{R\sim\mathcal{R}}[h(x', R) \neq y] \leq 1/2 - \eta$ and therefore by the Chernoff bound we have

$$\Pr_{R_1,R_2,\ldots,R_t}[h^{(R_1,R_2,\ldots,R_t)}(x') \neq y] \leq \Pr_{R_1,R_2,\ldots,R_t}[\sum_{i=1}^{t} \mathbb{1}[h(x, R_i) \neq y] \geq t/2] \leq \exp(-2\eta^2 t).$$

Finally, by a union bound over the set $\mathcal{A}(x)$ we have

$$\Pr_{R_1,R_2,\ldots,R_t}[\exists x' \in \mathcal{A}(x)\colon h^{(R_1,R_2,\ldots,R_t)}(x') \neq y] \leq |\mathcal{A}(x)| \cdot \exp(-2\eta^2 t) \leq \delta^2.$$

Since, this is true for any fixed (x, y) such that $\varepsilon(x, y) \leq 1/2 - \eta$, we have

$$\mathop{\mathbb{E}}_{\substack{(x,y)\sim D\\ R_1,R_2,\ldots,R_t}}[\exists x' \in \mathcal{A}(x)\colon h^{(R_1,R_2,\ldots,R_t)}(x') \neq y | \varepsilon(x, y) < 1/2 - \eta] \leq \delta^2,$$

and by Markov's inequality this yields

$$\Pr_{R_1,R_2\ldots,R_t}\left[\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[\exists x' \in \mathcal{A}(x)\colon h^{(R_1,R_2,\ldots,R_t)}(x') \neq y | \varepsilon(x, y) < 1/2 - \eta] \geq \delta\right] \leq \delta.$$

Our desired claim follows by noting that when (x, y) is sampled from $\mathcal{D}$, then with probability at most $\varepsilon(\eta)$ we have $\varepsilon(x, y) > 1/2 - \eta$:

$$\Pr_{R_1,R_2\ldots,R_t}\left[\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[\exists x' \in \mathcal{A}(x)\colon h^{(R_1,R_2,\ldots,R_t)}(x') \neq y] \geq \varepsilon(\eta) + \delta\right] \leq$$
$$\Pr_{R_1,R_2\ldots,R_t}\left[\varepsilon(\eta) + \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[\exists x' \in \mathcal{A}(x)\colon h^{(R_1,R_2,\ldots,R_t)}(x') \neq y | \varepsilon(x, y) < 1/2 - \eta] \geq \varepsilon(\eta) + \delta\right] \leq$$
$$\Pr_{R_1,R_2\ldots,R_t}\left[\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[\exists x' \in \mathcal{A}(x)\colon h^{(R_1,R_2,\ldots,R_t)}(x') \neq y | \varepsilon(x, y) < 1/2 - \eta] \geq \delta\right] \leq \delta.$$

□

C.8 Proof of Theorem 3.2

Proof. Fix a sample (x, y) such that $\gamma(\rho, x, y) \leq 1/2 - \eta$. We will show that

$$\Pr_{R_1,\ldots,R_t}\left[\exists x' \in \mathcal{A}(x) : \frac{\rho^{(R_1,R_2,\ldots,R_t)}(x')}{\textsc{Robust}(h, x')} \notin [1 - \beta, 1 + \alpha]\right] \leq \delta.$$

For this consider a fixed $x' \in \mathcal{A}(x)$. Notice that

$$\Pr_{R\sim\mathcal{R}}\left[\frac{\rho(x', R)}{\textsc{Robust}(h, x')} \notin [1 - \beta, 1 + \alpha]\right] \leq 1/2 - \eta.$$

Therefore, by the Chernoff bound we have

$$\Pr_{R_1,\ldots,R_t}\left[\frac{\rho^{(R_1,R_2,\ldots,R_t)}(x')}{\textsc{Robust}(h, x')} \notin [1 - \beta, 1 + \alpha]\right] \leq$$
$$\Pr_{R_1,\ldots,R_t}\left[\sum_{i=1}^{t} \mathbb{1}[\frac{\rho(x', R_i)}{\textsc{Robust}(h, x')} \leq 1 - \beta] \geq t/2\right] + \Pr_{R_1,\ldots,R_t}\left[\sum_{i=1}^{t} \mathbb{1}[\frac{\rho(x', R_i)}{\textsc{Robust}(h, x')} \geq 1 + \alpha] \geq t/2\right] \leq 2 \cdot \exp\left(-2\eta^2 t\right).$$

Finally, by a union bound over $\mathcal{A}(x)$ we have

$$\Pr_{R_1,\ldots,R_t}\left[\exists x' \in \mathcal{A}(x) : \frac{\rho^{(R_1,R_2,\ldots,R_t)}(x')}{\textsc{Robust}(h, x')} \notin [1 - \beta, 1 + \alpha]\right] \leq 2|\mathcal{A}(x)| \cdot \exp\left(-2\eta^2 t\right) \leq \delta^2.$$

Since, this is true for any fixed (x, y) such that $\varepsilon(x, y) \leq 1/2 - \eta$, we have

$$\mathop{\mathbb{E}}_{\substack{(x,y)\sim D\\ R_1,R_2,\ldots,R_t}}[\exists x' \in \mathcal{A}(x)\colon \frac{\rho^{(R_1,R_2,\ldots,R_t)}(x')}{\textsc{Robust}(h, x')} \notin [1 - \beta, 1 + \alpha] | \varepsilon(x, y) < 1/2 - \eta] \leq \delta^2,$$

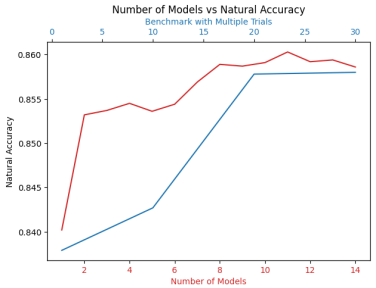
and by Markov's inequality this yields

$$\Pr_{R_1,R_2\ldots,R_t}\left[\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[\exists x' \in \mathcal{A}(x)\colon \frac{\rho^{(R_1,R_2,\ldots,R_t)}(x')}{\textsc{Robust}(h, x')} \notin [1 - \beta, 1 + \alpha] | \varepsilon(x, y) < 1/2 - \eta] \geq \delta\right] \leq \delta.$$

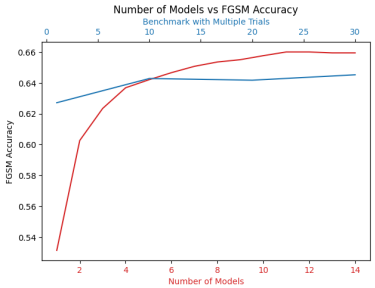
Our desired claim follows by noting that when (x, y) is sampled from $\mathcal{D}$, then with probability at most $\varepsilon(\eta)$ we have $\varepsilon(x, y) > 1/2 - \eta$:

$$\Pr_{R_1,R_2\ldots,R_t}\left[\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[\exists x' \in \mathcal{A}(x)\colon \frac{\rho^{(R_1,R_2,\ldots,R_t)}(x')}{\textsc{Robust}(h, x')} \notin [1 - \beta, 1 + \alpha]] \geq \varepsilon(\eta) + \delta\right] \leq$$
$$\Pr_{R_1,R_2\ldots,R_t}\left[\varepsilon(\eta) + \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[\exists x' \in \mathcal{A}(x)\colon \frac{\rho^{(R_1,R_2,\ldots,R_t)}(x')}{\textsc{Robust}(h, x')} \notin [1 - \beta, 1 + \alpha] | \varepsilon(x, y) < 1/2 - \eta] \geq \varepsilon(\eta) + \delta\right] \leq$$
$$\Pr_{R_1,R_2\ldots,R_t}\left[\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[\exists x' \in \mathcal{A}(x)\colon \frac{\rho^{(R_1,R_2,\ldots,R_t)}(x')}{\textsc{Robust}(h, x')} \notin [1 - \beta, 1 + \alpha] | \varepsilon(x, y) < 1/2 - \eta] \geq \delta\right] \leq \delta.$$

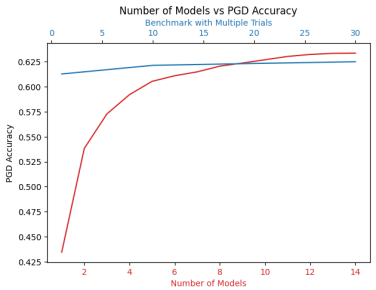
□



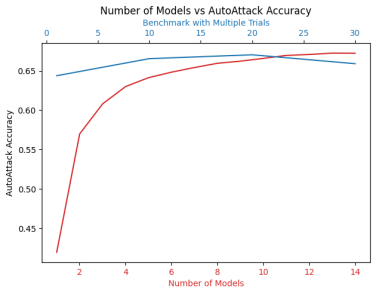
(a) Natural Accuracy



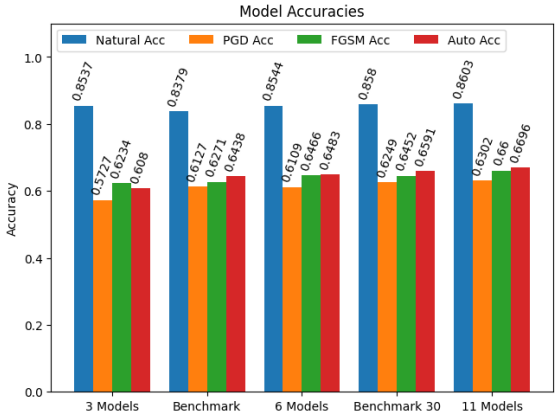
(b) FGSM



(c) PGD



(d) Auto Attack



(e) Bar graph comparing our model against the benchmark for various adversaries: blocks 1, 3 and 5 correspond to the number of derandomized models used, where as blocks 2 and 5 correspond to the number of trails by the benchmark model

Figure 1: Results of derandomizing the Random Projection Filters framework of Dong and Xu [2023] using Theorem 3.1. The red and blue curves correspond to the derandomized model and the benchmark respectively.