Projected Randomized Smoothing for Certified Adversarial Robustness

Samuel Pfrommer, Brendon G. Anderson, Somayeh Sojoudi
Department of Electrical Engineering and Computer Sciences
University of California, Berkeley
Berkeley, CA 94720
sam.pfrommer@berkeley.edu, bganderson@berkeley.edu, sojoudi@berkeley.edu

Abstract

Randomized smoothing is the current state-of-the-art method for producing provably robust classifiers. While randomized smoothing typically yields robust ℓ_2-ball certificates, recent research has generalized provable robustness to different norm balls as well as anisotropic regions. This work considers a classifier architecture that first projects onto a low-dimensional approximation of the data manifold and then applies a standard classifier. By performing randomized smoothing in the low-dimensional projected space, we characterize the certified region of our smoothed composite classifier back in the high-dimensional input space and prove a tractable lower bound on its volume. We show experimentally on CIFAR-10 and SVHN that classifiers without the initial projection are vulnerable to perturbations that are normal to the data manifold yet are captured by the certified regions of our method. We compare the volume of our certified regions against various baselines and show that our method improves on the state-of-the-art by many orders of magnitude.

1 Introduction

Despite their state-of-the-art performance on a variety of machine learning tasks, neural networks are vulnerable to adversarial inputs—inputs with small (often human-imperceptible) noise that is maliciously crafted to cause the algorithm to fail [8, 41, 35]. This sensitive behavior is unacceptable in contemporary safety-critical applications of neural networks, such as autonomous driving [9, 49] and the operations of power systems [21]. The works [15] and [29] highlight the validity and eminence of these threats, wherein both physical and digital adversarial perturbations are shown to cause image classification models to misclassify vehicle traffic signs.

Heuristics have been proposed to defend against various adversarial attacks, only to be defeated by stronger attack methods, leading to an "arms race" in the literature [10, 24, 5, 44, 32]. This has motivated researchers to consider certifiable robustness—theoretical proof that models perform reliably when subject to arbitrary attacks of a bounded norm [48, 47, 36, 4, 31]. Randomized smoothing, popularized in [25, 28, 12], remains one of the state-of-the-art methods for generating classifiers with certified robustness guarantees. Instead of directly classifying a given input, randomized smoothing intentionally corrupts the input with random noise and returns the most probable class, which, intuitively, "averages out" any potential adversarial perturbations in the data.

The seminal work [12] certifies that no adversarial perturbation within a certain ℓ_2-ball can cause the misclassification of a smoothed model using isotropic Gaussian noise of a fixed variance. Recent works have attempted to certify larger regions of the input space by turning to randomized smoothing with optimized variances [52], input-dependent variances [2, 46], and anisotropic distributions [13]. However, for a fixed variance, the certified radius is upper-bounded by a constant in the dimension d of the input [23], implying that the volume of the certified ℓ_2-ball degrades factorially fast as

Preprint. Under review.

$O(K^d\Gamma(\frac{d}{2}+1)^{-1})$, where Γ is Euler's gamma function and K is some positive constant [17]. Current input-dependent and anisotropic smoothing approaches have similarly been shown to suffer from the curse of dimensionality [40].

The small certified regions of randomized smoothing in high dimensions corroborate empirical findings that show increased robustness when precomposing classifiers with dimensionality reduction methods, e.g., principal component analysis projections [7] and autoencoders [37]. These findings align with the manifold hypothesis, which posits that real datasets lie on a low-dimensional manifold in a high-dimensional feature space [16], and the related results showing that perturbation directions most useful to an adversary are the ones normal to this manifold [19, 54]. Thus, projecting inputs onto the manifold, or at least a low-dimensional subspace containing the manifold, should increase classification robustness. Methods taking this approach, such as [33] and [1], have worked well as heuristics, but lack theoretical robustness guarantees. Motivated by these works, we aim to enlarge the certifiably robust regions of randomized smoothing by performing the smoothing in a low-dimensional space in which adversarial access to the data's statistically insignificant yet vulnerable features has been eliminated.

1.1 Contributions

We propose *projected randomized smoothing*, whereby inputs are projected onto a low-dimensional linear subspace in which randomized smoothing is applied before classification. Our method combines the empirical successes of dimension-reducing projection methods with the theoretical guarantees of randomized smoothing to achieve the following contributions:

1. We theoretically characterize the geometry of the certified region in the (high-dimensional) input space and prove a tractable lower bound on the volume of this certified region.
2. We empirically demonstrate that classifiers can be attacked along subspaces spanned by statistically insignificant features that contribute nothing to classification accuracy, which are vulnerabilities that projected randomized smoothing certifiably eliminates.
3. Experiments on CIFAR-10 [22] and SVHN [34] show that our method yields certified regions with order-of-magnitude larger volumes than prior state-of-the-art smoothing schemes.

1.2 Related works

Robustification via dimensionality reduction. The work [7] shows that linearly projecting inputs onto the top principal components of the training data before classification is an effective defense method. The authors of [37] nonlinearly preprocess test data using denoising and dimension-reducing autoencoders, and find a substantial increase in classification accuracy when the inputs are subject to the popular fast gradient sign method attack. In [33], the authors use super-resolution to project images onto the natural data manifold and obtain high empirical robustness for convolutional neural networks. The work [1] shows that decreasing the codimension of data, i.e., decreasing the difference between the intrinsic dimension of the data manifold and the dimension of the input space in which it is embedded, generally leads to increased robustness of models defined on that input space. The work [51] precomposes classifiers with orthogonal encoders and performs randomized smoothing in the encoder's low-dimensional latent space as a means to speed up the sample-based smoothing procedure. To the best of our knowledge, [51] is the only work that provides certified robustness guarantees for models using dimensionality reduction at the input—all of the other referenced works are heuristic—and their choice of orthogonal encoders ensures that the certified ℓ_2-ball in the input space has the same radius as that in the latent space. On the other hand, the method we propose uses a robustification-motivated projection for which we prove more general (anisotropic) certified regions.

Certification via randomized smoothing. The work [12] develops randomized smoothing using an isotropic Gaussian distribution with input-independent variance to obtain certified ℓ_2-balls. A subsequent line of works attempts to generalize randomized smoothing to other classes of certified regions, e.g., Wasserstein, "ℓ_0"-, ℓ_1-, and ℓ_∞-balls [27, 26, 43, 50]. Various approaches have been taken to enlarge the certified regions. For example, [38] unifies adversarial training with randomized smoothing to obtain state-of-the-art certified ℓ_2-radii. The authors of [52] incorporate the certified ℓ_2-radius into the model's training objective as a means to enlarge certified regions. The method in [53] optimizes over base classifiers to increase the size of more general ℓ_p-balls. Recent works

have also considered optimizing the certified region pointwise in the input space, but generally these methods require locally constant smoothing distributions to ensure that the resulting certificates are mathematically valid [2, 46, 40, 3]. To further strengthen the robustness guarantees of randomized smoothing, the recent works [13, 14, 42] have turned to certifying anisotropic regions of the input space. For example, [13] maximizes the volume of certified ellipsoids and generalized cross-polytopes of the form $\{x \in \mathbb{R}^d : \|Ax\|_p \leq b\}$ for $p \in \{1, 2\}$, allowing for the certification of perturbations that are potentially larger in magnitude than the minimum adversarial perturbation. We show in Section 4 that our proposed method is able to outperform these methods by leveraging dimensionality reduction. We emphasize that volume (Lebesgue measure) is the natural scalar measure for size of anisotropic certified regions of the input space and is the standard notion considered by prior works [30, 13, 42].

1.3 Notation

We denote the set of real numbers by $\mathbb{R}$. The ℓ_2-norm of a vector $x \in \mathbb{R}^n$ is denoted by $\|x\|$, whereas the general ℓ_p-norm is given an explicit subscript $\|x\|_p$. The range and nullspace of a matrix $U \in \mathbb{R}^{m \times n}$ are denoted by $\mathcal{R}(U) \subseteq \mathbb{R}^m$ and $\mathcal{N}(U) \subseteq \mathbb{R}^n$, respectively. The $n \times n$ identity matrix is written as I_n. For a random variable X with distribution $\mathcal{D}$ and a measurable function f, the expectation of $f(X)$ is denoted by $\mathbb{E}_{X \sim \mathcal{D}} f(X)$. The multivariate normal distribution with mean $\mu \in \mathbb{R}^n$ and covariance $\Sigma \in \mathbb{R}^{n \times n}$ is given by $N(\mu, \Sigma)$. The cardinality of a set S is written as $|S|$. For a Lebesgue-measurable set $S \subseteq \mathbb{R}^n$ contained in a k-dimensional affine subspace, we write $\mathcal{V}_k(S)$, termed the k-dimensional volume of S, to mean the Lebesgue measure of S within that affine subspace. For sets $S, T \subseteq \mathbb{R}^n$, we denote their Minkowski sum by $S + T = \{x + y : x \in S,\ y \in T\}$. Euler's gamma function is denoted by Γ. Recall that $\Gamma(n) = (n-1)!$ when n is a positive integer.

2 Classifier architecture

Consider the task of classifying inputs from a zero-centered cube $C^d = [-1/2, 1/2]^d \subseteq \mathbb{R}^d$ into c distinct classes $\mathcal{Y} = \{1, 2, \ldots, c\}$.[1] Under the randomized smoothing framework, we begin with a given classifier $f_\theta \colon \mathbb{R}^d \to [0, 1]^c$, parameterized by θ, that maps into the probability simplex over c classes. The problem at hand is to increase the robustness of f_θ with certifiable guarantees.

Vanilla randomized smoothing. We give a brief overview of how this would be accomplished using vanilla randomized smoothing [12]. Randomized smoothing takes the *base classifier* f_θ and smooths it with Gaussian noise on the input to yield the associated smoothed soft and hard classifiers

$$f^s(x) = \mathop{\mathbb{E}}_{\epsilon \sim N(0, \sigma^2 I_d)} f_\theta(x + \epsilon), \quad g(x) = \arg\max_{y \in \mathcal{Y}} f^s(x)_y,$$

where $f^s(x)_y$ denotes the yth component of the vector $f^s(x)$ and σ is a hyperparameter. [12, Theorem 1] then gives, under certain conditions, a certified ℓ_2 ball for a particular input $x \in \mathbb{R}^d$; namely, that $g(x + \delta) = g(x)$ for all $\|\delta\| < R$, where $R > 0$ is determined by the confidence of the smoothed classifier at x. We leverage this result for our approach and refer interested readers to [12] for additional details on the computation of the smoothing expectation and precise formula for R.

Projected randomized smoothing. Motivated by the relationships between robustness and dimensionality described in Section 1, we consider $p < d$ and let $P \colon \mathbb{R}^d \to \mathbb{R}^p$ be a projection into $\mathbb{R}^p$ defined by $P(x) = U^\intercal x$, where $U \in \mathbb{R}^{d \times p}$ is a semi-orthogonal matrix satisfying $U^\intercal U = I_p$. Similarly, we let the reconstruction $\tilde{P} \colon \mathbb{R}^p \to \mathbb{R}^d$ be defined by $\tilde{P}(\tilde{x}) = U\tilde{x}$. Throughout, we let $v_1, \ldots, v_{d-p} \in \mathbb{R}^d$ be an orthonormal basis for $\mathcal{N}(U^\intercal)$ and let $v_{d-p+1}, \ldots, v_d \in \mathbb{R}^d$ denote the orthonormal columns of U. In practice, we will instantiate the columns of U as the first p principal components of a random subset of the training dataset. With the dimension-reducing projection P in place, we consider the classifier architecture consisting of the composition

$$f = f_\theta \circ \tilde{P} \circ P.$$

In particular, f first uses P to project inputs into the low-dimensional space $\mathbb{R}^p$ and then reconstructs the inputs in a lossy way using $\tilde{P}$ before feeding them through the classifier f_θ. We generally finetune f_θ to account for the slight image corruption associated with the projection step.

[1] The zero-centered cube is used without loss of generality instead of $[0, 1]^d$ for notational convenience and compatibility with results from the mathematical literature.

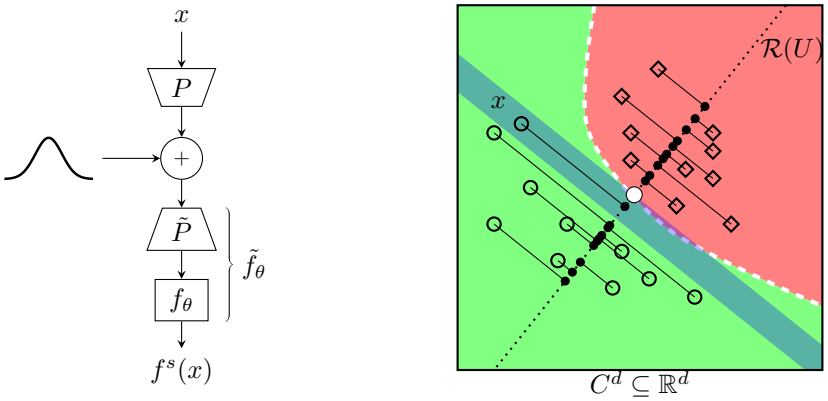

Figure 1: (a) Projected randomized smoothing architecture. Inputs x are projected into low-dimensional space by P, smoothed with Gaussian noise, and then reconstructed by $\tilde{P}$ and classified by f_θ. (b) Illustration of projected randomized smoothing. The white circle represents the smoothed decision boundary in $\mathbb{R}^p$. The blue area represents the certified region around x in $\mathbb{R}^d$ of projected randomized smoothing classifier g.

We now propose *projected randomized smoothing*, wherein randomized smoothing is performed in the compressed space $\mathbb{R}^p$. To do so, we define $\tilde{f}_\theta \colon \mathbb{R}^p \to [0,1]^c$ by $\tilde{f}_\theta = f_\theta \circ \tilde{P}$ so that $f = \tilde{f}_\theta \circ P$, and we smooth $\tilde{f}_\theta$ by adding Gaussian noise in its low-dimensional input space to obtain a new classifier $\tilde{f}^s_\theta \colon \mathbb{R}^p \to [0,1]^c$ defined by

$$\tilde{f}^s_\theta(\tilde{x}) = \mathop{\mathbb{E}}_{\epsilon \sim N(0,\sigma^2 I_p)} \tilde{f}_\theta(\tilde{x} + \epsilon). \tag{1}$$

The new overall smoothed soft classifier is then given by

$$f^s = \tilde{f}^s_\theta \circ P, \tag{2}$$

and its structure is illustrated in Figure 1a. The corresponding hard classifier is then given by the $\arg\max$ of the soft classifier:[2]

$$g(x) = \arg\max_{y \in \mathcal{Y}} f^s(x)_y. \tag{3}$$

A graphical illustration of our approach for $d = 2$ is shown in Figure 1b. To summarize, classifying an input $x \in \mathbb{R}^d$ using projected randomized smoothing amounts to applying the mapping $x \mapsto g(x)$ defined by (1) through (3), and it is for g that we seek to derive certified regions of the input space.

3 Theoretical robustness certificates

In this section, we construct certified regions for g around arbitrary inputs x in the high-dimensional space $\mathbb{R}^d$. The key idea is that $\tilde{f}^s_\theta$ is ℓ_2-ball robust in the low-dimensional space $\mathbb{R}^p$, and the preimage of this ball in the original input space is then "large" as it includes the inputs in $\mathcal{N}(U^\intercal)$. We formalize the geometry of the certified region in Section 3.1 and introduce our metric of interest as the volume of the certified region restricted to the unit cube of feasible inputs. In Section 3.2, we provide a lower bound on this volume in high-dimensional spaces that involves solving an ℓ_∞-norm linear regression. Section 3.3 compares the asymptotic behavior of the volume of the certified region of g with the standard ℓ_2-ball certificates as the input dimension grows large. Finally, we discuss runtime and limitations in Section 3.4. For ease of exposition, all proofs are deferred to the appendices.

3.1 Characterizing the certified region geometry

In the following two propositions, we characterize the geometry of the projected randomized smoothing classifier g in the high-dimensional input space $\mathbb{R}^d$ based on the certified ℓ_2-robustness of the classifier $\tilde{f}^s_\theta$ in the low-dimensional projected space $\mathbb{R}^p$.

[2]For ease of exposition, we assume throughout that all $\arg\max$ yield singleton sets and therefore equality signs may be used unambiguously.

Definition 1. Let $\tilde{x} \in \mathbb{R}^p$ and $R \geq 0$. The classifier $\tilde{f}_\theta^s \colon \mathbb{R}^p \to [0,1]^c$ is said to be *certified at $\tilde{x}$ with radius R* if

$$\arg\max_{y \in \mathcal{Y}} \tilde{f}_\theta^s(\tilde{x} + \tilde{\delta})_y = \arg\max_{y \in \mathcal{Y}} \tilde{f}_\theta^s(\tilde{x})_y$$

for all $\tilde{\delta} \in \mathbb{R}^p$ satisfying $\|\tilde{\delta}\| \leq R$.

Proposition 1. *Let $x \in \mathbb{R}^d$ and $R \geq 0$. If $\tilde{f}_\theta^s$ is certified at $P(x) = U^\intercal x$ with radius R, then $g(x+\delta) = g(x)$ for all $\delta \in \Delta^U(R) \subseteq \mathbb{R}^d$, where*

$$\Delta^U(R) := \{\delta \in \mathbb{R}^d : \|U^\intercal \delta\| \leq R\}.$$

Proposition 2. *Let $R \geq 0$. The certified region $\Delta^U(R)$ can be expressed as the Minkowski sum*

$$\Delta^U(R) = B_p^U(R) + \mathcal{N}(U^\intercal),$$

where $B_p^U(R) \subseteq \mathbb{R}^d$ is a p-dimensional ball embedded into $\mathcal{R}(U)$:

$$B_p^U(R) := \left\{ \sum_{i=d-p+1}^{d} \beta_{i-d+p} v_i : \|\beta\| \leq R,\ \beta \in \mathbb{R}^p \right\}.$$

Propositions 1 and 2 characterize the geometry of the certified region of our classifier g. Proposition 1 provides an easy-to-check condition for a perturbation $\delta \in \mathbb{R}^d$ to lie in the certified region, while Proposition 2 formalizes the same geometry as a hypercylinder consisting of a low-dimensional sphere that is "extruded" along the nullspace of the projection P, allowing us to certify adversarial off-manifold inputs of potentially very large magnitude that are projected back onto the natural data manifold. Intuitively, the certified region $\Delta^U(R)$ is potentially much larger than an ℓ_2-ball of radius R in $\mathbb{R}^d$, as it captures perturbations in the nullspace of $U^\intercal$ whose dimensionality is large in a compressed scenario where $p \ll d$.

3.2 Lower-bounding the certified region volume

To compare a standard ℓ_2-ball certificate with our certified region $\Delta^U(R)$, which does not immediately come equipped with a notion of "radius," we adopt the perspective of recent works, e.g., [30, 13, 42], by considering our metric of interest to be the volume of the certified region. One immediate issue is that the volume of $\Delta^U(R)$ is infinite since $\mathcal{N}(U^\intercal)$ is an unbounded subspace. To enable meaningful comparisons, we restrict ourselves to measuring the volume of $\Delta^U(R)$ contained in the cube $C^d = [-1/2, 1/2]^d$ of possible inputs. This amounts to computing the volume

$$\mathbb{V}_d\left(C^d \cap \Delta_x^U(R)\right), \tag{4}$$

where we recall that $\mathbb{V}_d$ measures d-dimensional volume in Euclidean space, and

$$\Delta_x^U(R) := \{x + \delta : \delta \in \Delta^U(R)\},$$

with R chosen such that $\tilde{f}_\theta^s$ is certified at $P(x)$ with radius R so that $g(x') = g(x)$ for all $x' \in \Delta_x^U(R)$ by Proposition 1. Computing the volume in (4) is highly nontrivial, especially in high-dimensional input spaces. Instead, we develop a tractable lower bound on $\mathbb{V}_d(C^d \cap \Delta_x^U(R))$ throughout the remainder of this section. Since $\Delta_x^U(R)$ contains affine subspaces, this derivation rests heavily on theory regarding cube-subspace intersections in high dimensions. The most important result for our purposes comes from [45], which showed the following.

Theorem 1 ([45])**.** *Let S_k be a k-dimensional linear subspace of $\mathbb{R}^d$. Then $\mathbb{V}_k(C^d \cap S_k) \geq 1$.*

This result proved Good's conjecture and generalized a previous result for the $k = d-1$ case [18]. We begin with an extension of Theorem 1 to cubes of non-unit side length, and then to intersections with affine subspaces which do not necessarily contain the origin.

Corollary 1. *Let S_k be a k-dimensional linear subspace of $\mathbb{R}^d$ and rC^d be a zero-centered cube of side length $r > 0$. Then $\mathbb{V}_k(rC^d \cap S_k) \geq r^k$.*

Algorithm 1 Prediction and certification

def functions PREDICT, CERTIFY as in [12]

function PROJECTPREDICT(f_θ, U, σ, x, n, α)
def $P(x) = U^\intercal x$, $\tilde{P}(\tilde{x}) = U\tilde{x}$
return PREDICT($f_\theta \circ \tilde{P}, \sigma, P(x), n, \alpha$)

function PROJECTCERTIFY(f_θ, U, σ, x, n_0, n, α)
def $P(x) = U^\intercal x$, $\tilde{P}(\tilde{x}) = U\tilde{x}$, $(d, p) \leftarrow \text{shape}(U)$
ABSTAIN, $\hat{c}_A, R =$ CERTIFY($f_\theta \circ \tilde{P}, \sigma, P(x), n_0, n, \alpha$)
if ABSTAIN **then return** ABSTAIN
compute an orthonormal basis $v_1, \ldots, v_{d-p}$ for $\mathcal{N}(U^\intercal)$
solve the optimization

$$t \leftarrow \min_{\alpha \in \mathbb{R}^{d-p}} \left\| x + \textstyle\sum_{i=1}^{d-p} \alpha_i v_i \right\|_\infty \tag{Alg1.1}$$

assign $R \leftarrow \min\{R, p(1-2t)/(2d)\}$
compute the certified volume lower bound $V \leftarrow \frac{\pi^{p/2}}{\Gamma(\frac{p}{2}+1)} R^p (1-2R-2t)^{d-p}$
return prediction $\hat{c}_A$ and volume bound V

Corollary 2. *Let $x \in \mathbb{R}^d$ and let $S_k(x) \subseteq \mathbb{R}^d$ be the k-dimensional affine subspace $S_k(x) = \left\{ x + \sum_{i=1}^k \alpha_i v_i : \alpha \in \mathbb{R}^k \right\}$ spanned by arbitrary vectors $v_1, \ldots, v_k$ and passing through x. Let $t \geq 0$ be the minimal ℓ_∞-norm of a point in $S_k(x)$:*

$$t := \inf_{x' \in S_k(x)} \|x'\|_\infty = \inf_{\alpha \in \mathbb{R}^k} \left\| x + \sum_{i=1}^k \alpha_i v_i \right\|_\infty . \tag{5}$$

Then, for all $r > 2t$, it holds that

$$\mathbb{V}_k(rC^d \cap S_k(x)) \geq (r-2t)^k.$$

Corollary 2 generalizes Corollary 1 to arbitrary affine subspaces. Note that in the case where $S_k(x)$ contains the origin, $t = 0$ and the bound from Corollary 1 is recovered. We are now ready to present the main result of this section.

Theorem 2. *Let $x \in C^d$, let t be defined as in* (5) *with $k = d - p$, and let $R \in [0, 1/2 - t]$. If $\tilde{f}_\theta^s$ is certified at $P(x) = U^\intercal x$ with radius R, then*

$$\mathbb{V}_d(C^d \cap \Delta_x^U(R)) \geq \frac{\pi^{p/2}}{\Gamma(\frac{p}{2}+1)} R^p (1-2R-2t)^{d-p}. \tag{6}$$

Notice that the lower bound given in Theorem 2 does not monotonically increase with the certified radius R from the randomized smoothing performed in $\mathbb{R}^p$. Therefore, if the certified radius R is large enough, we may be able to improve our lower bound on the volume $\mathbb{V}_d(C^d \cap \Delta_x^U(R))$ by using a smaller certified radius (which is of course still valid), and in particular, we may choose the optimal such radius to use according to the following closed-form expression.

Proposition 3. *Let t and R be as in Theorem 2. The optimization problem*

$$\sup_{r \in [0,R]} \frac{\pi^{p/2}}{\Gamma\left(\frac{p}{2}+1\right)} r^p \left(1-2r-2t\right)^{d-p} \tag{7}$$

is solved by

$$r^* = \min\left\{ R, \frac{p(1-2t)}{2d} \right\}.$$

The overall certification procedure derived in this section is summarized in Algorithm 1.

3.3 Asymptotic behavior of the certified volume lower bound

We briefly compare the volume lower bound (6) of the projected randomized smoothing certified region to that of a standard certified ℓ_2-ball. The volume of a d-dimensional ℓ_2-ball $B_d(R) \coloneqq \{x \in \mathbb{R}^d : \|x\| \leq R\}$ of radius $R \geq 0$ is well-known (e.g., see [17, Theorem 2.44, Corollary 2.55]) to be

$$\mathcal{V}_d\left(B_d(R)\right) = \frac{\pi^{d/2}}{\Gamma(\frac{d}{2}+1)} R^d.$$

While the numerator scales exponentially in d, the denominator $\Gamma(\frac{d}{2}+1)$ scales factorially, leading to tiny ℓ_2-ball certified volumes in high-dimensional input spaces. By contrast, the denominator in our bound (6) scales factorially in the *projected dimension* p, where $p \ll d$. This suggests dramatic improvements in the volume of our certified regions, dominating any exponentially-scaled concessions in the numerator of (6). We verify our analysis experimentally in Section 4.2.

3.4 Runtime and limitations

Our certification strategy has two additional computational steps outside of the PREDICT and CERTIFY subroutines from the conventional randomized smoothing method of [12]. The first is a one-time computation of the principal components of the data that occurs at the beginning of training. The second is computing the ℓ_∞-regression in (Alg1.1), which we solve as a linear program using the standard epigraph formulation. For the CIFAR-10 and SVHN datasets considered in this work, the added runtime is comparable to the certification sampling step from [12]. Namely, we found that the ℓ_∞-regression averaged around 16 seconds for CIFAR-10 and 19 seconds for SVHN.[3]

The number of variables and constraints in the optimization (Alg1.1) scales linearly with $d - p$. Since generally $p \ll d$, this makes the volume approximation of the certified region computationally intensive in high-dimensional input spaces. We remark that it is still trivial to check whether any particular perturbation lies in the certified region using Proposition 1—it is just that computing a lower bound on the volume of this region for comparison purposes becomes more challenging. For a natural image dataset such as ImageNet, the analysis of Section 3.3 suggests that the certified region volume improvements would in fact be substantially larger than those for CIFAR-10. Computational verification of this fact would likely leverage techniques from the large-scale ℓ_∞-regression literature, e.g., [39], and is outside the scope of this work.

4 Experiments

This section reports our experiments on the CIFAR-10 dataset, and to save space we defer the results for SVHN to Appendix B.3. We first demonstrate in Section 4.1 that networks are vulnerable to ℓ_∞-bounded attacks in the subspace of low-variance principal components, to which our architecture is provably robust. Section 4.2 then presents results comparing the volume of the projected randomized smoothing certified regions to a variety of baseline certified classifiers.

4.1 Networks are vulnerable to low-variance PCA attacks

Consider adversarial perturbations $\delta \in \mathcal{N}(U^\intercal)$ contained in the span of a dataset's low-variance principal components, where here we take U to contain sufficient components to account for 99% of the dataset variance. Such a perturbation is known to be essentially orthogonal to the true data manifold, and therefore it is reasonable to expect a truly robust classifier to be invariant to small perturbations in $\mathcal{N}(U^\intercal)$.

Our method is directly robust to such perturbations under the simple condition that we use fewer components in our initial projection step, as demonstrated in Proposition 2. We now investigate whether this adds a degree of robustness over a typical neural network classifier. The answer is affirmative. Namely, we show that our subspace attack can attain a comparable attack success rate to a standard ℓ_∞-bounded projected gradient descent (PGD) attack, with roughly a four-fold increase in the size of the admissible ℓ_∞-ball.

[3] All experiments were run on a Ubuntu 20.04 virtual machine with 6 VCPUs, 56 GiB RAM, and a Tesla K80 GPU. Complete reproduction takes roughly 0.06 GPU years.

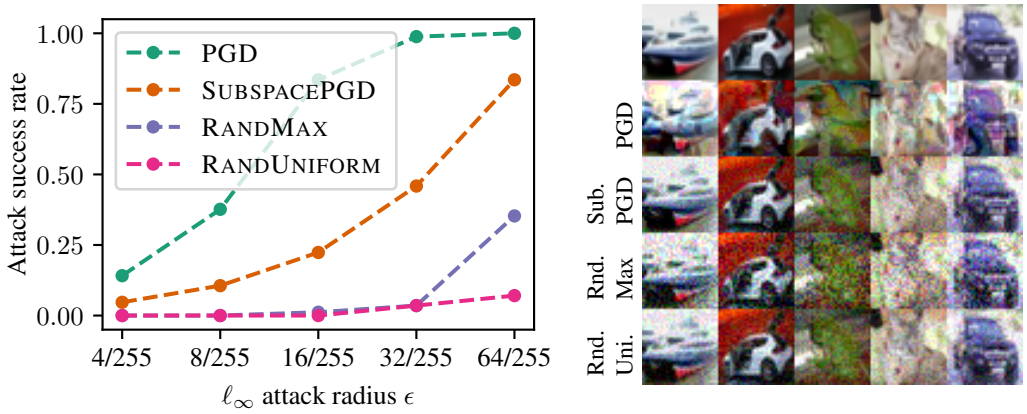

Figure 2: (a) Adversarial attack success rates for the PGD, SUBSPACEPGD, and random attack strategies. (b) Perturbation examples for CIFAR-10 with an attack radius of $\epsilon = 32/255$. The top row represents the original image.

Formally, consider a particular hard classifier g, to which we assume that our adversaries have white-box access, and take a specific input x that g classifies correctly. We first consider the standard projected gradient descent attack strategy $\text{PGD}(x, \epsilon)$ which seeks to construct a perturbation $\|\delta\|_\infty \leq \epsilon$ such that $x + \delta \in C^d$ and $g(x + \delta) \neq g(x)$. As $x + \delta \in C^d$ if and only if $\|x + \delta\|_\infty \leq 1/2$, satisfying both ℓ_∞-norm constraints on δ is easily accomplished using clipping. Our routine $\text{SUBSPACEPGD}(x, \epsilon)$ adds the additional constraint $\delta \in \mathcal{N}(U^\intercal)$. Note that finding a perturbation that satisfies $\delta \in \mathcal{N}(U^\intercal)$, $\|\delta\|_\infty \leq \epsilon$, and $x + \delta \in C^d$ is nontrivial, as projection onto one set generally removes an input from the other set. The precise details of our attack strategy are detailed in Appendix B.1. For reference, we also consider RANDMAX and RANDUNIFORM, which generate perturbations randomly on the boundary of and uniformly in the attack ℓ_∞ ball, respectively. We instantiate g as the Wide ResNet considered in [50] with the default hyperparameters and $\sigma = 0.15$ Gaussian noise augmentation during training. See Appendix B.4.1 for the attack hyperparameters.

Figure 2 demonstrates that unprotected classifiers are indeed vulnerable to adversarial perturbations in the subspace of low-variance principal components. Enlargements of the attack radius magnitude do not invalidate that these are true adversarial attacks, as the perturbed images in the third row of Figure 2b are still easily classified by a human. Furthermore, SUBSPACEPGD adversarial examples are substantially less perceptible than PGD attacks of the same magnitude, which tend to produce stronger visual distortions of the image: take as a representative example the region to the left of the frog's head in the second row of the center column in Figure 2b. This is likely because the PGD attacks have access to high-variance principal components which convey the true information content of the dataset. Despite visually appearing random, we establish in Figure 2a that the SUBSPACEPGD attack is significantly more successful than true random-noise attacks of the same magnitude.

These results suggest that undefended classifiers can be attacked in the subspace of low-variance principal components, to which projected randomized smoothing is provably robust by Proposition 2.

4.2 Certified region comparison

Having established that the certified region of projected randomized smoothing provides a meaningful robustness improvement against low-variance principal component attacks, we now compare the volume of our certified region with several baselines. Namely, we evaluate the ℓ_2-balls of [12] (denoted RS), the ℓ_1- and ℓ_∞-balls of [50] (denoted $\text{RS4A} - \ell_1$ and $\text{RS4A} - \ell_\infty$, respectively), and the anisotropic ellipsoids of [13] (denoted ANCER), without use of the associated memory module.

Some additional remarks on the inclusion of [13] are warranted. As noted in [40], without the inclusion of the memory module, the local certificate optimization technique in [13] yields overly optimistic and mathematically incorrect certificates as the smoothing distribution varies between inputs. The work [13] corrects this with the use of a memory module that records previous inputs to ensure compatibility of the smoothing certificates. However, this results in a classifier that is dependent on the input order and adds ambiguity about what classifier is actually being certified, as the smoothed classifier is modified at test time after each input. We therefore discard the memory

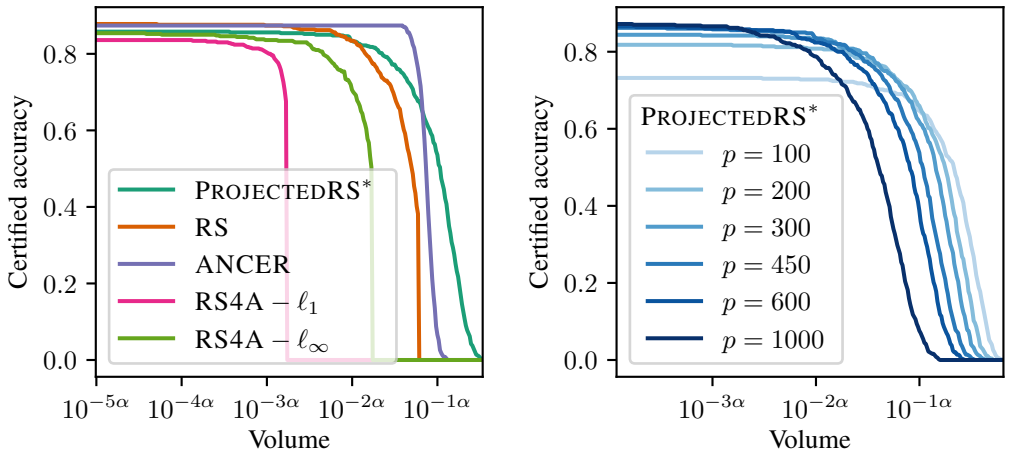

Figure 3: (a) Certified region volumes for CIFAR-10, with our method highlighted by an asterisk. Here $\alpha \approx 3465$ is a scaling constant corresponding to the d-dimensional unit ball volume; i.e. $\forall_d(B_d(1)) = 10^{-\alpha}$. (b) Certified region volumes while varying the projected space dimension p.

module and report the certified volume at each point as if the locally optimized smoothing distribution were being used globally. This yields an upper bound on the certified volume of any data-dependent anisotropic ellipsoidal smoothing method and is thus a very strong baseline to compare against.

Our CIFAR-10 results are summarized in Figure 3 and Table 1 in the appendix. We achieve state-of-the-art median certified volumes, easily outperforming standard randomized smoothing and even the optimistic ANCER baseline by 706 and 2453 *orders of magnitude* on CIFAR-10 and SVHN, respectively. The larger improvement on SVHN is attributable to the higher compressibility of the dataset, as we mention in Appendix B.3. Figure 4 in Appendix B.2 suggests that our performance derives from the added robustness of our method against low-variance features, as the radii of the projected-space certified balls are similar to those of standard randomized smoothing in the original input space. This further validates the asymptotic dimension analysis in Section 3.3. Note that although the ANCER baseline is able to achieve higher accuracy at smaller volumes, its certificates are mathematically invalid [40], and our method significantly outperforms ANCER at larger volumes.

Figure 3b examines the certified accuracy curves over a range of choices for the dimensionality p of the compressed space. For large p, image reconstruction is near-perfect as $p = 620$ covers 99% of variance in the CIFAR-10 dataset. Thus, methods with $p \geq 300$ have comparable accuracy at small regions, with the certified volumes increasing as the dimensionality of the projected space decreases, corroborating the discussion in Section 3.3. We are therefore able to increase the robustness of our classifier to disturbances that are normal to the manifold with only a 2% drop in accuracy (Table 1).

5 Conclusion

Motivated by the manifold hypothesis, we consider a classifier architecture that first projects onto a principal component approximation of the data manifold and then applies randomized smoothing in the low-dimensional projected space. This yields a precise characterization of the input-space certified region as capturing disturbances in the projection nullspace. We interpret this as a certifiable robustification against vulnerable features that are irrelevant to the dataset information content as they are normal to the data manifold. We show that unprotected classifiers, unlike our method, are vulnerable to such perturbations by explicitly constructing adversarial examples in the span of the low-variance principal components. We prove a volumetric lower bound on the intersection of our certified region with the unit cube of feasible inputs and derive additional ways to tighten the bound.

Comparing against state-of-the-art ℓ_1, ℓ_2, ℓ_∞, and anisotropic baselines shows that our classifier produces certified regions with many orders of magnitude greater volume. This confirms an asymptotic analysis that shows that our method's certified volumes decay factorially in the low dimension of the *projected space*, while competing methods decay factorially in the high dimension of the *input space*. Future research directions include examining more sophisticated dimensionality reduction techniques while maintaining certified guarantees in the original input space.

References

[1] Sheila Alemany and Niki Pissinou. The dilemma between data transformations and adversarial robustness for time series application systems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022.

[2] Motasem Alfarra, Adel Bibi, Philip H.S. Torr, and Bernard Ghanem. Data dependent randomized smoothing. *arXiv preprint arXiv:2012.04351*, 2020.

[3] Brendon G. Anderson and Somayeh Sojoudi. Certified robustness via locally biased randomized smoothing. In *Learning for Dynamics and Control*. PMLR, 2022.

[4] Brendon G. Anderson, Ziye Ma, Jingqi Li, and Somayeh Sojoudi. Tightened convex relaxations for neural network robustness certification. In *Proceedings of the 59th IEEE Conference on Decision and Control*, 2020.

[5] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *International conference on machine learning*, pages 274–283. PMLR, 2018.

[6] Dimitri Pi. Bertsekas. *Nonlinear Programming*. Athena Scientific, third edition, 2016.

[7] Arjun Nitin Bhagoji, Daniel Cullina, Chawin Sitawarin, and Prateek Mittal. Enhancing robustness of machine learning systems via data transformations. In *2018 52nd Annual Conference on Information Sciences and Systems (CISS)*, pages 1–5. IEEE, 2018.

[8] Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Šrndić, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. Evasion attacks against machine learning at test time. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 387–402. Springer, 2013.

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

[10] Nicholas Carlini and David 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.

[11] Hongyan Chang, Ta Duy Nguyen, Sasi Kumar Murakonda, Ehsan Kazemi, and Reza Shokri. On adversarial bias and the robustness of fair machine learning. *arXiv preprint arXiv:2006.08669*, 2020.

[12] Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *International Conference on Machine Learning*, pages 1310–1320. PMLR, 2019.

[13] Francisco Eiras, Motasem Alfarra, M Pawan Kumar, Philip HS Torr, Puneet K Dokania, Bernard Ghanem, and Adel Bibi. Ancer: Anisotropic certification via sample-wise volume maximization. *arXiv preprint arXiv:2107.04570*, 2021.

[14] Ecenaz Erdemir, Jeffrey Bickford, Luca Melis, and Sergul Aydore. Adversarial robustness with non-uniform perturbations. *Advances in Neural Information Processing Systems*, 34, 2021.

[15] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. Robust physical-world attacks on deep learning visual classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1625–1634, 2018.

[16] Charles Fefferman, Sanjoy Mitter, and Hariharan Narayanan. Testing the manifold hypothesis. *Journal of the American Mathematical Society*, 29(4):983–1049, 2016.

[17] Gerald B Folland. *Real analysis: Modern techniques and their applications*. John Wiley & Sons, second edition, 1999.

[18] Douglas Hensley. Slicing the cube in r^n and probability (bounds for the measure of a central cube slice in r^n by probability methods). *Proceedings of the American Mathematical Society*, 73(1):95–100, 1979. ISSN 00029939, 10886826. URL `http://www.jstor.org/stable/2042889`.

[19] Susmit Jha, Uyeong Jang, Somesh Jha, and Brian Jalaian. Detecting adversarial examples using data manifolds. In *MILCOM 2018-2018 IEEE Military Communications Conference (MILCOM)*, pages 547–552. IEEE, 2018.

[20] Hoki Kim. Torchattacks: A pytorch repository for adversarial attacks. *arXiv preprint arXiv:2010.01950*, 2020.

[21] Weicong Kong, Zhao Yang Dong, Youwei Jia, David J Hill, Yan Xu, and Yuan Zhang. Short-term residential load forecasting based on lstm recurrent neural network. *IEEE Transactions on Smart Grid*, 10(1):841–851, 2017.

[22] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[23] Aounon Kumar, Alexander Levine, Tom Goldstein, and Soheil Feizi. Curse of dimensionality on randomized smoothing for certifiable robustness. In *International Conference on Machine Learning*, pages 5458–5467. PMLR, 2020.

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

[25] Mathias Lecuyer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, and Suman Jana. Certified robustness to adversarial examples with differential privacy. In *IEEE Symposium on Security and Privacy*, pages 656–672. IEEE, 2019.

[26] Guang-He Lee, Yang Yuan, Shiyu Chang, and Tommi Jaakkola. Tight certificates of adversarial robustness for randomly smoothed classifiers. *Advances in Neural Information Processing Systems*, 32, 2019.

[27] Alexander Levine and Soheil Feizi. Wasserstein smoothing: Certified robustness against Wasserstein adversarial attacks. In *International Conference on Artificial Intelligence and Statistics*, pages 3938–3947. PMLR, 2020.

[28] Bai Li, Changyou Chen, Wenlin Wang, and Lawrence Carin. Certified adversarial robustness with additive noise. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[29] Aishan Liu, Xianglong Liu, Jiaxin Fan, Yuqing Ma, Anlan Zhang, Huiyuan Xie, and Dacheng Tao. Perceptual-sensitive gan for generating adversarial patches. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 1028–1035, 2019.

[30] Chen Liu, Ryota Tomioka, and Volkan Cevher. On certifying non-uniform bounds against adversarial attacks. In *International Conference on Machine Learning*, pages 4072–4081. PMLR, 2019.

[31] Ziye Ma and Somayeh Sojoudi. A sequential framework towards an exact SDP verification of neural networks. In *2021 IEEE 8th International Conference on Data Science and Advanced Analytics (DSAA)*, pages 1–8. IEEE, 2021.

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

[33] Aamir Mustafa, Salman H Khan, Munawar Hayat, Jianbing Shen, and Ling Shao. Image super-resolution as a defense against adversarial attacks. *IEEE Transactions on Image Processing*, 29: 1711–1724, 2019.

[34] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011.

[35] Anh Nguyen, Jason Yosinski, and Jeff Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 427–436, 2015.

[36] Aditi Raghunathan, Jacob Steinhardt, and Percy S Liang. Semidefinite relaxations for certifying robustness to adversarial examples. In *Advances in Neural Information Processing Systems*, pages 10877–10887, 2018.

[37] Rajeev Sahay, Rehana Mahfuz, and Aly El Gamal. Combatting adversarial attacks through denoising and dimensionality reduction: A cascaded autoencoder approach. In *2019 53rd Annual conference on information sciences and systems (CISS)*, pages 1–6. IEEE, 2019.

[38] Hadi Salman, Jerry Li, Ilya Razenshteyn, Pengchuan Zhang, Huan Zhang, Sebastien Bubeck, and Greg Yang. Provably robust deep learning via adversarially trained smoothed classifiers. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[39] Fumin Shen, Chunhua Shen, Rhys Hill, Anton van den Hengel, and Zhenmin Tang. Fast approximate ℓ_∞ minimization: Speeding up robust regression. *Computational Statistics & Data Analysis*, 77:25–37, 2014.

[40] Peter Súkeník, Aleksei Kuvshinov, and Stephan Günnemann. Intriguing properties of input-dependent randomized smoothing. *arXiv preprint arXiv:2110.05365*, 2021.

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

[42] Lucas Matthew Tecot. Robustness verification with non-uniform randomized smoothing. Master's thesis, University of California, Los Angeles, 2021.

[43] Jiaye Teng, Guang-He Lee, and Yang Yuan. ℓ_1 adversarial robustness certificates: A randomized smoothing approach. *Preprint*, 2020. URL `https://openreview.net/forum?id=H1lQIgrFDS`.

[44] Jonathan Uesato, Brendan O'donoghue, Pushmeet Kohli, and Aaron Oord. Adversarial risk and the dangers of evaluating against weak attacks. In *International Conference on Machine Learning*, pages 5025–5034. PMLR, 2018.

[45] Jeffrey Vaaler. A geometric inequality with applications to linear forms. *Pacific Journal of Mathematics*, 83(2):543–553, 1979.

[46] Lei Wang, Runtian Zhai, Di He, Liwei Wang, and Li Jian. Pretrain-to-finetune adversarial training via sample-wise randomized smoothing. *Preprint*, 2021. URL `https://openreview.net/pdf?id=Te1aZ2myPIu`.

[47] Lily Weng, Huan Zhang, Hongge Chen, Zhao Song, Cho-Jui Hsieh, Luca Daniel, Duane Boning, and Inderjit Dhillon. Towards fast computation of certified robustness for ReLU networks. In *International Conference on Machine Learning*, pages 5276–5285. PMLR, 2018.

[48] Eric Wong and Zico Kolter. Provable defenses against adversarial examples via the convex outer adversarial polytope. In *International Conference on Machine Learning*, pages 5286–5295. PMLR, 2018.

[49] Bichen Wu, Forrest Iandola, Peter H. Jin, and Kurt Keutzer. SqueezeDet: Unified, small, low power fully convolutional neural networks for real-time object detection for autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 129–137, 2017.

[50] Greg Yang, Tony Duan, J Edward Hu, Hadi Salman, Ilya Razenshteyn, and Jerry Li. Randomized smoothing of all shapes and sizes. In *International Conference on Machine Learning*, pages 10693–10705. PMLR, 2020.

[51] Huimin Zeng, Jiahao Su, and Furong Huang. Certified defense via latent space randomized smoothing with orthogonal encoders. *arXiv preprint arXiv:2108.00491*, 2021.

[52] Runtian Zhai, Chen Dan, Di He, Huan Zhang, Boqing Gong, Pradeep Ravikumar, Cho-Jui Hsieh, and Liwei Wang. Macer: Attack-free and scalable robust training via maximizing certified radius. In *International Conference on Learning Representations*, 2020.

[53] Dinghuai Zhang, Mao Ye, Chengyue Gong, Zhanxing Zhu, and Qiang Liu. Black-box certification with randomized smoothing: A functional optimization based framework. In *Advances in Neural Information Processing Systems*, volume 33, pages 2316–2326, 2020.

[54] Yonggang Zhang, Xinmei Tian, Ya Li, Xinchao Wang, and Dacheng Tao. Principal component adversarial example. *IEEE Transactions on Image Processing*, 29:4804–4815, 2020.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] We claim three main contributions: characterization of the certified region geometry (Section 3.1) and a lower bound on its volume (Section 3.2); vulnerability of undefended classifiers to low-variance component attacks (Section 4.1); and larger certified region volumes (Section 4.2).
   (b) Did you describe the limitations of your work? [Yes] See Section 3.4.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See Appendix C.1.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] No assumptions are required on the neural network classifier f_θ besides that it outputs to a probability simplex, and all other assumptions are reproduced in the relevant theorem statement.
   (b) Did you include complete proofs of all theoretical results? [Yes] See Appendix A.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See supplemental material.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section B.4.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A] These are not appropriate for our experiments.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See the footnote for Section 3.4.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] We use pretrained models from [50] and the CIFAR-10 [22] and SVHN [34] datasets, which are cited in the introduction.
   (b) Did you mention the license of the assets? [Yes] See Appendix B.5.
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] See supplemental material.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Supplementary material for Section 3 (Theoretical robustness certificates)

Proposition 1. *Let $x \in \mathbb{R}^d$ and $R \geq 0$. If $\tilde{f}_\theta^s$ is certified at $P(x) = U^\intercal x$ with radius R, then $g(x+\delta) = g(x)$ for all $\delta \in \Delta^U(R) \subseteq \mathbb{R}^d$, where*

$$\Delta^U(R) := \{\delta \in \mathbb{R}^d : \|U^\intercal \delta\| \leq R\}.$$

Proof. Let $\delta \in \Delta^U(R)$. Then

$$g(x+\delta) = \arg\max_{y \in \mathcal{Y}} \tilde{f}_\theta^s(P(x+\delta))_y = \arg\max_{y \in \mathcal{Y}} \tilde{f}_\theta^s(P(x) + U^\intercal \delta)_y.$$

Since $\|U^\intercal \delta\| \leq R$ by definition of $\Delta^U(R)$ and $\tilde{f}_\theta^s$ is certified at $P(x)$ with radius R, we have that

$$g(x+\delta) = \arg\max_{y \in \mathcal{Y}} \tilde{f}_\theta^s(P(x))_y = g(x).$$

□

Proposition 2. *Let $R \geq 0$. The certified region $\Delta^U(R)$ can be expressed as the Minkowski sum*

$$\Delta^U(R) = B_p^U(R) + \mathcal{N}(U^\intercal),$$

where $B_p^U(R) \subseteq \mathbb{R}^d$ is a p-dimensional ball embedded into $\mathcal{R}(U)$:

$$B_p^U(R) := \left\{ \sum_{i=d-p+1}^{d} \beta_{i-d+p} v_i : \|\beta\| \leq R,\ \beta \in \mathbb{R}^p \right\}.$$

Proof. Let $y = y_1 + y_2$ with $y_1 \in B_p^U(R)$ and $y_2 \in \mathcal{N}(U^\intercal)$. Then

$$\|U^\intercal y\| = \|U^\intercal y_1\| = \|\beta\| \leq R,$$

so $y \in \Delta^U(R)$.

On the other hand, let $y \in \Delta^U(R)$ as defined in Proposition 1. We can decompose $y = y_1 + y_2$ for $y_1 \in \mathcal{R}(U)$ and $y_2 \in \mathcal{N}(U^\intercal)$. Then there exists $\beta \in \mathbb{R}^p$ such that $y_1 = U\beta = \sum_{i=d-p+1}^{n} \beta_{i-d+p} v_i$, so $\|U^\intercal y_1\| = \|\beta\|$ and therefore $\|\beta\| \leq R$. □

Corollary 1. *Let S_k be a k-dimensional linear subspace of $\mathbb{R}^d$ and rC^d be a zero-centered cube of side length $r > 0$. Then $\mathcal{V}_k(rC^d \cap S_k) \geq r^k$.*

Proof. Note that

$$\begin{aligned} rC^d \cap S_k &= \{x \in \mathbb{R}^d : \|x\|_\infty \leq r/2,\ x \in S_k\} \\ &= \{rx \in \mathbb{R}^d : \|rx\|_\infty \leq r/2,\ rx \in S_k\} \\ &= \{rx \in \mathbb{R}^d : \|x\|_\infty \leq 1/2,\ x \in S_k\}, \end{aligned}$$

since $x \in S_k$ if and only if $rx \in S_k$, by linearity of S_k. This is now equivalent to the set $r(C^d \cap S_k)$, and we have scaled our k-dimensional subset by a uniform factor r. Therefore, $\mathcal{V}_k(rC^d \cap S_k) = \mathcal{V}_k(r(C^d \cap S_k)) = r^k \mathcal{V}_k(C^d \cap S_k)$ by [17, Theorem 2.44]. Thus, by Theorem 1, we have $\mathcal{V}_k(rC^d \cap S_k) \geq r^k$. □

Corollary 2. *Let $x \in \mathbb{R}^d$ and let $S_k(x) \subseteq \mathbb{R}^d$ be the k-dimensional affine subspace $S_k(x) = \left\{x + \sum_{i=1}^k \alpha_i v_i : \alpha \in \mathbb{R}^k\right\}$ spanned by arbitrary vectors $v_1, \ldots, v_k$ and passing through x. Let $t \geq 0$ be the minimal ℓ_∞-norm of a point in $S_k(x)$:*

$$t := \inf_{x' \in S_k(x)} \|x'\|_\infty = \inf_{\alpha \in \mathbb{R}^k} \left\| x + \sum_{i=1}^{k} \alpha_i v_i \right\|_\infty. \tag{5}$$

Then, for all $r > 2t$, it holds that

$$\mathcal{V}_k(rC^d \cap S_k(x)) \geq (r - 2t)^k.$$

Proof. First, notice that the infimum in (5) is attained since $\|\cdot\|_\infty$ is continuous and coercive, and $S_k(x)$ is closed in the standard topology on $\mathbb{R}^d$ [6]. Let $x^* \in S_k(x)$ be a point that attains the infimum in (5) so that $\|x^*\|_\infty = t$. If $r > 2t$, then x^* is contained in the interior of rC^d. In this case, we can construct a nonempty cube centered at x^* with side lengths $r - 2t > 0$ that is contained in rC^d. Now, the plane $S_k(x)$ passes through x^*, and therefore Corollary 1 yields the result since volume is preserved under translation [17, Theorem 2.42]. □

Theorem 2. *Let $x \in C^d$, let t be defined as in* (5) *with $k = d - p$, and let $R \in [0, 1/2 - t]$. If $\tilde{f}_\theta^s$ is certified at $P(x) = U^\intercal x$ with radius R, then*

$$\mathcal{V}_d(C^d \cap \Delta_x^U(R)) \geq \frac{\pi^{p/2}}{\Gamma(\frac{p}{2}+1)} R^p (1 - 2R - 2t)^{d-p}. \tag{6}$$

Proof. The characterization of $\Delta^U(R)$ in Proposition 2 yields

$$\Delta_x^U(R) = B_p^U(R) + S_{d-p}^{\mathcal{N}(U^\intercal)}(x),$$

where

$$S_{d-p}^{\mathcal{N}(U^\intercal)}(x) := \{x\} + \mathcal{N}(U^\intercal)$$

is the affine subspace of $\mathbb{R}^d$ spanned by $\mathcal{N}(U^\intercal)$ and passing through x, which has dimension $d - p$. Therefore, the following is an inner-approximation of $\Delta_x^U(R)$:

$$\tilde{\Delta}_x^U(R) := B_p^U(R) + \left((1-2R)C^d \cap S_{d-p}^{\mathcal{N}(U^\intercal)}(x)\right) \subseteq B_p^U(R) + S_{d-p}^{\mathcal{N}(U^\intercal)}(x) = \Delta_x^U(R).$$

If we can show that $\tilde{\Delta}_x^U(R) \subseteq C^d$, then $\tilde{\Delta}_x^U(R) \subseteq C^d \cap \Delta_x^U(R)$, in which case the volume of $\tilde{\Delta}_x^U(R)$ will lower-bound the volume of $C^d \cap \Delta_x^U(R)$. To prove that this holds, let $y = y_1 + y_2 \in \tilde{\Delta}_x^U(R)$ with $y_1 \in B_p^U(R)$ and $y_2 \in (1-2R)C^d \cap S_{d-p}^{\mathcal{N}(U^\intercal)}(x)$. Then

$$\|y\|_\infty \leq \|y_1\|_\infty + \|y_2\|_\infty \leq R + \frac{1-2R}{2} = \frac{1}{2},$$

by the fact that $\|y_1\|_\infty \leq \|y_1\| = \|U\beta\| = \|\beta\|$ for some $\beta \in \mathbb{R}^p$ with $\|\beta\| \leq R$ due to the semi-orthogonality of U, and by the fact that $y_2 \in (1-2R)C^d$. Therefore, indeed it holds that $\tilde{\Delta}_x^U(R) \subseteq C^d$. Thus, all that remains is to lower-bound $\mathcal{V}_d(\tilde{\Delta}_x^U(R))$. To this end, notice that $B_p^U(R) \subseteq \mathcal{R}(U)$ and $(1-2R)C^d \cap S_{d-p}^{\mathcal{N}(U^\intercal)}(x) \subseteq \{x\} + \mathcal{N}(U^\intercal)$, so $B_p^U(R)$ and $(1-2R)C^d \cap S_{d-p}^{\mathcal{N}(U^\intercal)}(x)$ are contained in orthogonal affine subspaces, and therefore $\mathcal{V}_d(\tilde{\Delta}_x^U(R)) = \mathcal{V}_p(B_p^U(R))\mathcal{V}_{d-p}((1-2R)C^d \cap S_{d-p}^{\mathcal{N}(U^\intercal)}(x))$. The p-dimensional volume of the embedded ball ℓ_2-ball $B_p^U(R)$ is well-known (e.g., see [17, Theorem 2.44, Corollary 2.55]) to be

$$\mathcal{V}_p(B_p^U(R)) = \frac{\pi^{p/2}}{\Gamma(\frac{p}{2}+1)} R^p.$$

On the other hand, since $2R < 1 - 2t$, it holds that $1 - 2R > 2t$. Hence Corollary 2 gives that the $(d-p)$-dimensional volume of $(1-2R)C^d \cap S_{d-p}^{\mathcal{N}(U^\intercal)}(x)$ is lower-bounded as

$$\mathcal{V}_{d-p}((1-2R)C^d \cap S_{d-p}^{\mathcal{N}(U^\intercal)}(x)) \geq (1 - 2R - 2t)^{d-p}.$$

Therefore,

$$\mathcal{V}_d(\tilde{\Delta}_x^U(R)) \geq \frac{\pi^{p/2}}{\Gamma(\frac{p}{2}+1)} R^p (1 - 2R - 2t)^{d-p},$$

which concludes the proof. □

Proposition 3. *Let t and R be as in Theorem 2. The optimization problem*

$$\sup_{r \in [0,R]} \frac{\pi^{p/2}}{\Gamma\left(\frac{p}{2}+1\right)} r^p \left(1 - 2r - 2t\right)^{d-p} \tag{7}$$

is solved by

$$r^* = \min\left\{R, \frac{p(1-2t)}{2d}\right\}.$$

Proof. It suffices to maximize $h(r) \coloneqq r^p \left(1 - 2r - 2t\right)^{d-p}$ over $r \in [0, R]$. The gradient of h vanishes at points satisfying

$$\begin{aligned}
\frac{dh}{dr}(r) &= pr^{p-1}\left(1 - 2r - 2t\right)^{d-p} - 2(d-p)r^p\left(1 - 2r - 2t\right)^{d-p-1} \\
&= r^{p-1}\left(1 - 2r - 2t\right)^{d-p-1}\left(p\left(1 - 2r - 2t\right) - 2(d-p)r\right) \\
&= r^{p-1}\left(1 - 2r - 2t\right)^{d-p-1}\left(p - 2pt - 2dr\right) \\
&= 0.
\end{aligned}$$

The set of all critical points satisfying this polynomial equation is $\left\{0, \frac{p(1-2t)}{2d}, 1/2 - t\right\}$. Notice that $0 < \frac{p(1-2t)}{2d} < \frac{p(1-2t)}{2p} = 1/2 - t$, and that $\frac{dh}{dr}(r) \geq 0$ for all $r \in \left[0, \frac{p(1-2t)}{2d}\right]$ whereas $\frac{dh}{dr}(r) \leq 0$ for all $r \in \left[\frac{p(1-2t)}{2d}, 1/2 - t\right]$. Hence, h is unimodal on $[0, 1/2 - t]$ with the maximizer $\frac{p(1-2t)}{2d}$. Therefore, if $R < \frac{p(1-2t)}{2d}$, then h is monotone increasing on the feasible interval $[0, R]$, which implies that the right endpoint $r^* = R$ is a maximizer of (7). On the other hand, if $R \geq \frac{p(1-2t)}{2d}$, then $\frac{p(1-2t)}{2d}$ is contained in the feasible interval $[0, R]$, and thus $r^* = \frac{p(1-2t)}{2d}$ is a maximizer of (7). $\square$

B Supplementary material for Section 4 (Experiments)

B.1 Subspace attack procedure

A typical PGD attack constructs adversarial examples by iteratively perturbing the image along the gradient of the loss and projecting onto the unit cube of feasible inputs:

$$x^{(i+1)} = P_{C^d}\left(x + P_{C_\epsilon^d}\left(\alpha \operatorname{sign}\left(\nabla_\delta \mathcal{L}(x^{(i)} + \delta, y) - x^{(i)}\right)\right)\right),$$

where $x^{(i)}$ is the ith iterate of the PGD attack, $\mathcal{L}$ is the loss function, $\operatorname{sign}(\cdot)$ is the element-wise sign operator, α is the step size hyperparameter, P_{C^d} projects a point in $\mathbb{R}^d$ onto C^d by simple clipping, and $P_{C_\epsilon^d}$ is defined similarly for the zero-centered cube of sidelength 2ϵ. We initialize $x^{(1)} = x$, where (x, y) are the original input and label from the dataset.

We desire a final perturbation δ such that $x + \delta \in C^d$, $\delta \in C_\epsilon^d$, and $\delta \in \mathcal{N}(U^\intercal)$. We first parameterize our perturbation in terms of the vectors $v_1, \ldots, v_{d-p}$ spanning $\mathcal{N}(U^\intercal)$. Stacking these vectors columnwise to yield $V \in \mathbb{R}^{d \times d-p}$, we can express our pertubation as $\delta = V\delta_V$ with $\delta_V \in \mathbb{R}^{d-p}$. We then iterate over our parameterized perturbations $\delta_V^{(i)}$, first solving for our "target" perturbation

$$\left(\delta_V^{(i+1)}\right)^* = \delta_V^{(i)} + \alpha \operatorname{sign}\left(\nabla_{\delta'} \mathcal{L}(x + V(\delta_V^{(i)} + \delta'), y)\right).$$

We then project the perturbation to satisfy the ℓ_∞-constraints, which takes the form of a quadratic program:

$$\begin{array}{ll}
\underset{\delta_V \in \mathbb{R}^{d-p}}{\text{minimize}} & \left\| V\left(\delta_V^{(i+1)}\right)^* + V\delta_V \right\|_2^2 \\
\text{subject to} & \|V\delta_V\|_\infty \leq \epsilon, \\
& \|x + V\delta_V\|_\infty \leq 1/2.
\end{array}$$

This program is always feasible with $\delta_V = 0$, and its solution satisfies our requirements for each iteration of the attack procedure.

B.2 CIFAR-10 additional results

We provide a quantitative interpretation of the data in Figure 3a in Table 1. The first column reports the smoothed classifier accuracy for each method, disregarding certified volume, while the second column reports the median certified volume for correctly classified samples. We use the median instead of the mean due to the $\log$-scaled nature of our data.

	Accuracy	Median cert. vol. ($\log_{10}$)
PROJECTEDRS	85.8%	$\mathbf{-3175}$
RS	**87.8%**	-4377
ANCER	87.4%	-3881
RS4A $-\ell_1$	83.8%	-9573
RS4A $-\ell_\infty$	85.4%	-6102

Table 1: CIFAR certification performance.

Here we present an additional plot comparing the radii of the *low-dimensional* projected randomized smoothing balls to the radii of standard randomized smoothing balls in the high-dimensional space. These are very similar, suggesting that the increase in the volume of the certified region comes from the "extrustion" of the ball, which amounts to added robustness against unnecessary features that are removed in the initial projection step. For the anisotropic ANCER method, we report the geometric mean of the radii along each coordinate axis, which [13] defines to be the "proxy radius." We only compare methods with ℓ_2-based certified regions in this plot.

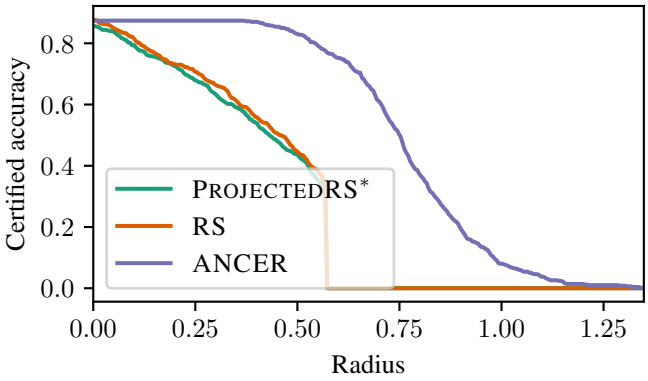

Figure 4: Certified radii on the CIFAR-10 dataset.

B.3 SVHN results

We reproduce the plots in Section 4 for the SVHN dataset. This dataset consists of digits collected from Google Street View house numbers. SVHN classifiers seem to be more robust to low-variance subspace attacks than CIFAR-10 classifiers, and Figure 5 we attack components accounting to the bottom 5% of variance (as opposed to 1% for CIFAR-10).

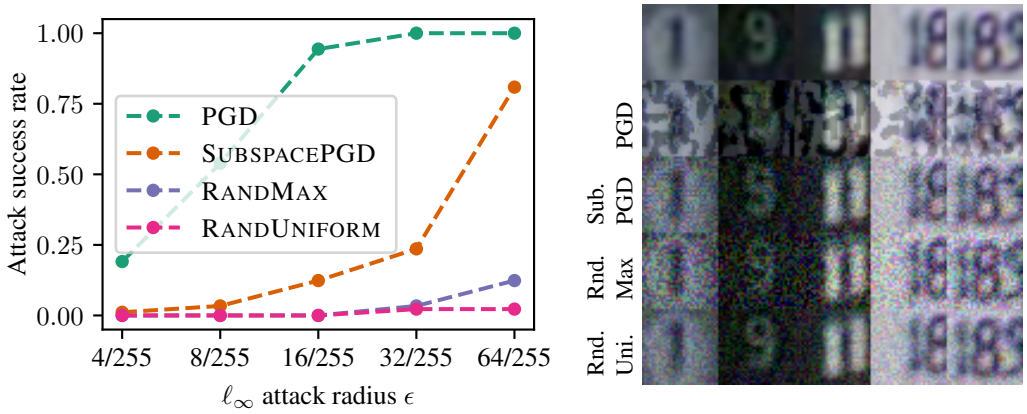

Figure 5: (a) Adversarial attack success rates for the PGD, SUBSPACEPGD, and random attack strategies on SVHN. (b) Perturbation examples for SVHN with an attack radius of $\epsilon = 32/255$. The top row represents the original image.

We reproduce the analog to Figure 3 below for SVHN.

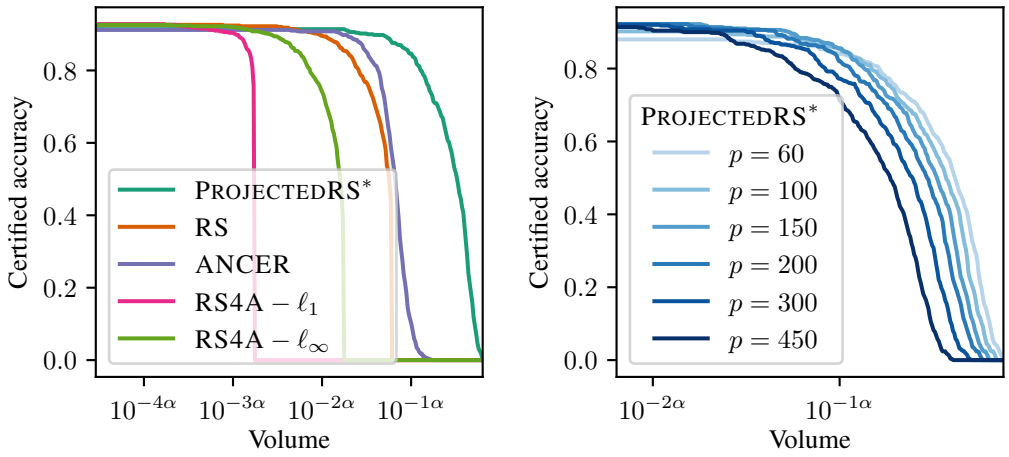

Figure 6: (a) Certified region volumes for SVHN, with our method highlighted by an asterisk. Here $\alpha \approx 3465$ is a scaling constant corresponding to the d-dimensional unit ball volume; i.e. $V_d(B_d(1)) = 10^{-\alpha}$. (b) Certified region volumes while varying the projected space dimension p.

Note that projected randomized smoothing significantly outperforms ANCER due to the higher compressibility of SVHN ($p = 150$). Finally, we report the analog of Table 1.

	Accuracy	Median cert. vol. ($\log_{10}$)
PROJECTEDRS	91.4%	$\mathbf{-1578}$
RS	92.6%	-4280
ANCER	91.2%	-4031
RS4A − ℓ_1	**93.0%**	-9573
RS4A − ℓ_∞	92.6%	-6171

Table 2: SVHN certification performance.

B.4 Hyperparameter selection

To maintain consistency, all networks we consider are Wide ResNets pretrained with various noise distributions using the code provided by [50]. For networks composed with an initial projection, we finetune the network with a learning rate or 0.001, momentum of 0.9, and weight decay of 0.0005 for 20 epochs, decaying the learning rate by a multiplicative factor of 0.95 per epoch.

B.4.1 Attack hyperparameters

We kept the attack hyperparameters fixed across both CIFAR-10 and SVHN. For the PGD attack, we use the torchattacks library with 40 steps and step size $\alpha = 2/255$ [20]. We lowered this to 5 steps with $\alpha = \epsilon/4$ for SUBSPACEPGD due to the solve time of the projection step.

B.4.2 CIFAR-10 certification hyperparameters

We include the results of our hyperparameter sweeps for CIFAR-10 in Figure 7. For the RS4A − ℓ_1 method, we used uniform noise and stability training to reproduce the state-of-the-art result from [50]. The RS4A − ℓ_∞ sweep used Guassian noise, which we found to perform better in practice. Our sweep over the ANCER learning rate held the number of steps and regularization weight fixed at their defaults of 900 and 2, respectively. All sweeps were performed over 500 random test samples besides ANCER which was run over 100 samples due to the method's high computational burden.

The results from these sweeps informed the choice of hyperparameters in Figure 3a. Namely, we choose $\sigma = 0.25$ for our RS4A − ℓ_1 baseline and $\sigma = 0.15$ for our RS4A − ℓ_∞ baseline, as the clean accuracy drops substantially for higher variances without approaching comparable certified

volume to the other methods considered. We choose a learning rate of 0.01 for ANCER and $p = 450$ components for projected randomized smoothing. All experiments in the hyperparameter sweeps were performed with the smoothing hyperparameters of $n_0 = 100$ samples to guess the smoothed class, $n = 10^4$ samples to lower-bound the smoothed class probability, and a confidence of $\alpha = 0.001$. For reproducing the final results in Figure 3a we increased n to 10^6 as is standard [12] and used 500 test samples to generate the plots. The attack experiment illustrated in Figure 2a was conducted over 100 test samples.

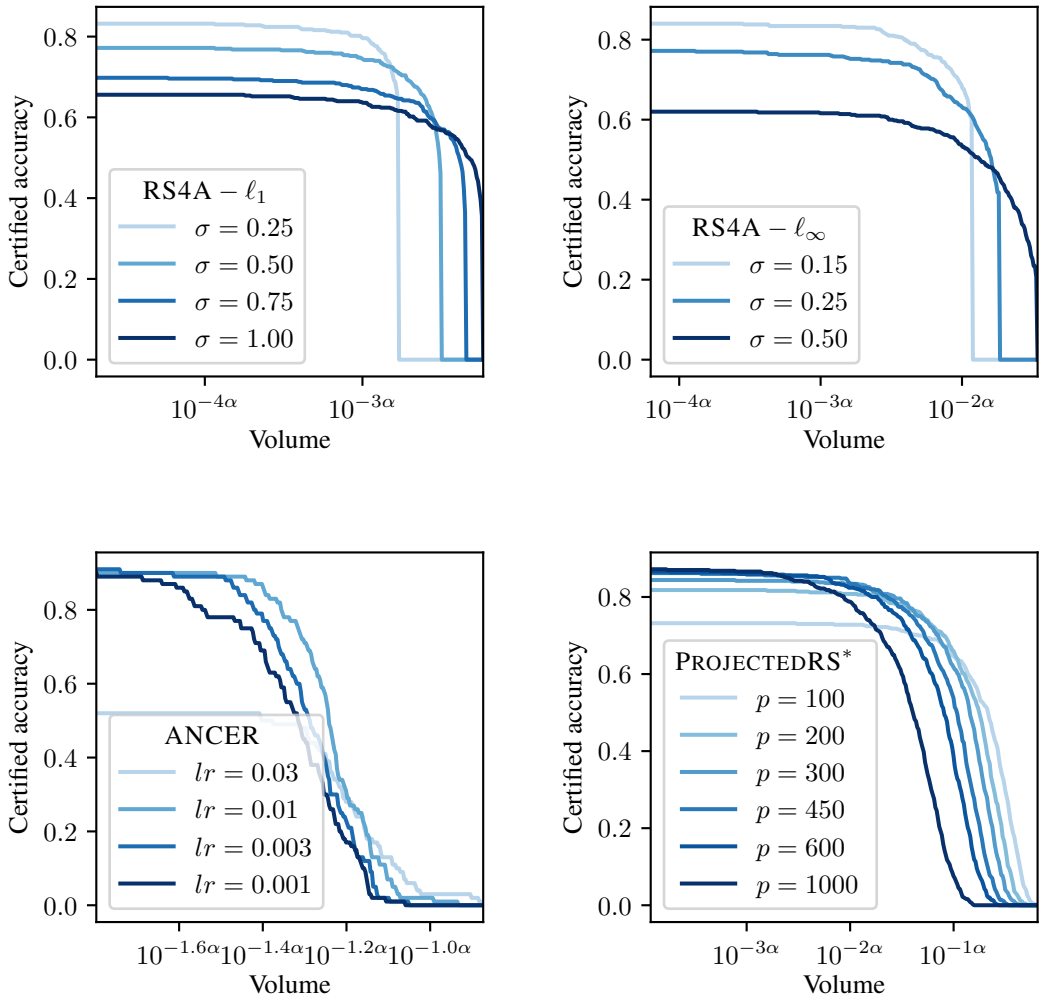

Figure 7: Hyperparameter sweeps for the CIFAR-10 dataset. Here $\alpha \approx 3465$ is a scaling constant corresponding to the d-dimensional unit ball volume; i.e. $V_d(B_d(1)) = 10^{-\alpha}$.

B.4.3 SVHN certification hyperparameters

The SVHN hyperparameter sweep is similar to that for CIFAR-10, besides the use of fewer principal components in the projected randomized smoothing sweeps due to the higher compressibility of the data. Our final plots in Figure 6a use $\sigma = 0.25$ for the ℓ_1 baseline, $\sigma = 0.15$ for the ℓ_∞ baseline, an ANCER learning rate of 0.01, and $p = 150$ for projected randomized smoothing.

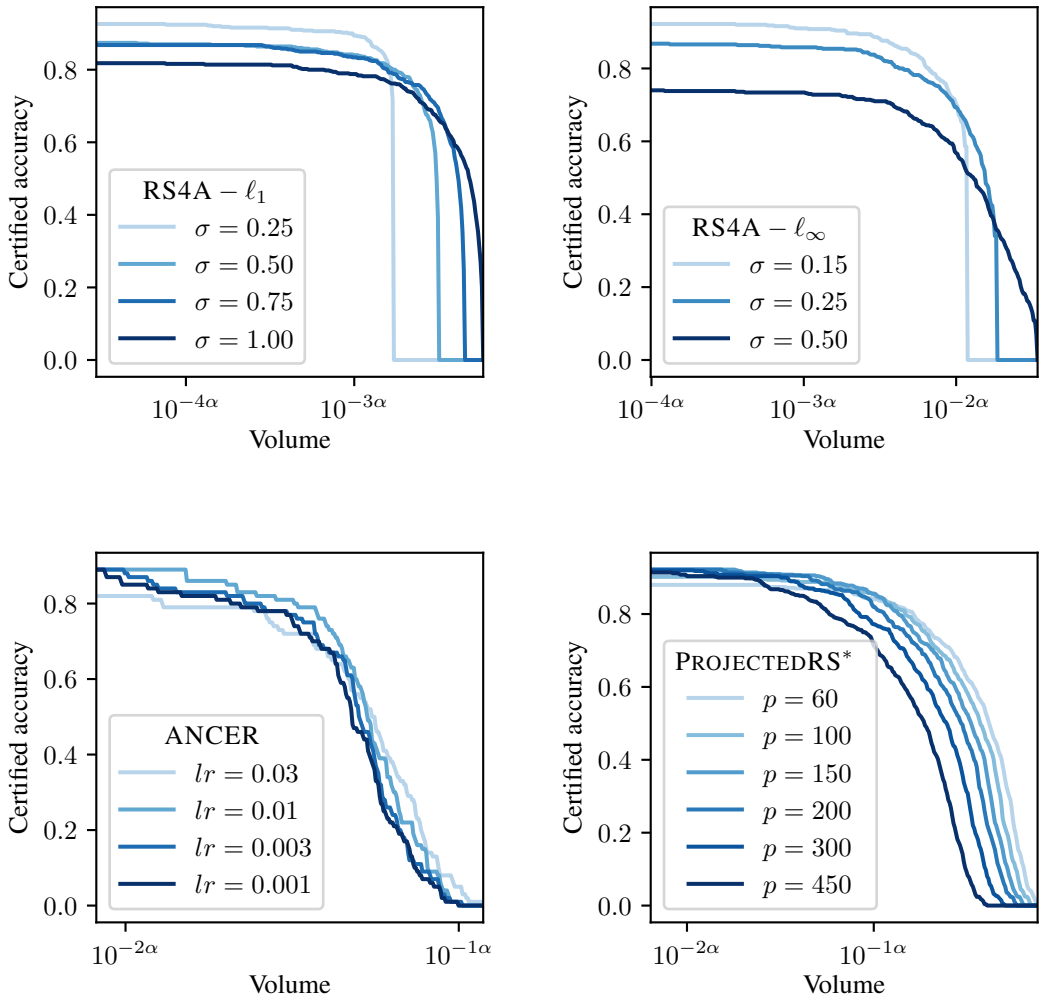

Figure 8: Hyperparameter sweeps for the SVHN dataset. Here $\alpha \approx 3465$ is a scaling constant corresponding to the d-dimensional unit ball volume; i.e. $\mathcal{V}_d(B_d(1)) = 10^{-\alpha}$.

B.5 Licenses

The CIFAR-10 dataset is covered by the MIT license, and the SVHN dataset is covered by the GPL 3 license.

C Supplementary material for Section 5 (Conclusion)

C.1 Societal impact

Improving neural network robustness is critical for ensuring that machine learning models are safe to deploy in real-world applications such as autonomous driving and medical diagnostics. However, potential side effects of improving robustness are not well understood, with some research suggesting that robust networks may be more biased [11]. While our work focuses on certifiable robustness, similar concerns may apply and are an important topic of future research.