Certifiably Robust Input-Dependent Randomized Smoothing via Lipschitz Standard Deviation Networks

Faith Bergstrom*
Mechanical Engineering Department
California Polytechnic State University
San Luis Obispo, CA 93407
fbergstr@calpoly.edu

Benjamin Sager*
Mathematics Department
California Polytechnic State University
San Luis Obispo, CA 93407
basager@calpoly.edu

Brendon G. Anderson
Mechanical Engineering Department
California Polytechnic State University
San Luis Obispo, CA 93407
bga@calpoly.edu

Abstract

We consider an input-dependent generalization of Gaussian randomized smoothing (RS) for constructing certifiably robust classifiers in which the variable smoothing distribution is parameterized as a neural network and is learned from data. In the case that the smoothing distribution is constructed to be zero-mean and isotropic for all inputs, we prove that the smoothed model satisfies a strong Lipschitz continuity property under mild regularity conditions that are easily enforced on the Gaussian's standard deviation network in practice. Using the determined Lipschitz constant, we derive a certified ℓ_2-radius of robustness for the smoothed model. Our Lipschitz constant and certified radius reduce to those of standard input-*independent* RS as a special case. Experiments on MNIST illustrate "best-of-both-worlds" performance; our input-dependent smoothing scheme simultaneously attains the high accuracy of small-variance standard RS on clean data and the high robustness of large-variance standard RS against adversarial attacks.

1 Introduction

Standard machine learning models are known to be highly sensitive to unreliable data. For example, classifiers have been shown to exhibit catastrophic failures when subjected to imperceptible adversarial attacks on their inputs [1, 2, 3]. This has motivated researchers to develop more robust ML models [4], as well as mathematical and computational frameworks for certifying model robustness [5, 6, 7].

Randomized smoothing (RS), popularized by Cohen et al. [8], Lecuyer et al. [9], Li et al. [10], is a post-processing method that certifiably robustifies a pretrained model by making predictions based on the average model output on inputs that are intentionally corrupted with Gaussian random noise. Despite remaining one of the state-of-the-art approaches for certified robustness, the standard formulation of RS is limited by an accuracy-robustness tradeoff [11, 12], as the technique amounts to smoothing the decision boundaries and consequently introduces underfitting when using large-variance noise [13, 14, 15]. Follow-up work has aimed to alleviate the tradeoff through a variety of approaches, including the incorporation of RS-friendly adversarial training and architecture modifications [16, 17, 18, 19], optimization of the smoothing noise distribution parameters [20, 15], and generalizing RS to utilize input-dependent smoothing noise (references discussed below). In

*Co-first authors and equal contribution. Order was determined by 2 out of 3 coin flips.

this work, we build on the input-dependent approach, which, despite gaining much interest in recent years, has seen limited progress thus far due to the significant mathematical challenges introduced when analyzing the robustness of models with variable smoothing distributions.

1.1 Related Works

Input-dependent smoothing is rooted in the idea that different data may benefit from different noise levels. Specifically, the prediction of data close to a classifier's decision boundary might be quite sensitive to the noising process, calling for low-magnitude, small-variance noise to maintain accuracy. Contrarily, data far from the decision boundary can withstand higher-variance noise while maintaining the correct prediction, which is consequently able to drown out stronger adversarial attacks.

The works Wang et al. [21], Eiras et al. [22], Alfarra et al. [23], Rumezhak et al. [24] have introduced various methods for implementing this input-dependent smoothing approach. However, their robustness certificates require the restrictive use of memory-based piecewise-constant smoothing parameters that rely on memorization of data that results in the model being dependent on the order that data is stored. The works Súkeník et al. [25], Anderson and Sojoudi [13] avoid this sensitivity on the data's order by using nearest-neighbor models to generate the input-dependent smoothing parameters, but these approaches also rely on storing data and constrain the smoothing scheme to adhere to a restrictive functional form. Chen et al. [26] avoids memory-based approaches and proposes to optimize the smoothing distribution's variance in a per-input fashion. However, their certified radii are mathematically invalid for the actual classifier used at test time [13]. Some key mathematical properties have been identified to rigorously implement input-dependent smoothing [27, 25], but the conditions tend to be restrictive, making the rigorous certification of practically useful radii remain an open challenge.

Other related works include the recent paper Lyu et al. [28], which proposes to adapt smoothing to each test-time input using a learned masking rule, but their approach does not explicitly vary the underlying input noise distribution parameters. Instead of varying the noise distribution across inputs, Chen et al. [29] proposes to vary the number of samples used in randomized smoothing's Monte Carlo estimates to optimize runtime on a per-input basis, and are therefore focused on an unrelated problem despite their method being referred to as "input-specific sampling."

1.2 Contributions

We achieve the following contributions in this paper:

1. Under easily-enforced, mild technical assumptions, we prove for the first time in the input-dependent smoothing literature a Lipschitz continuity property taking the same strong functional form as standard input-*independent* RS, i.e., $\Phi^{-1} \circ \overline{g}$ being Lipschitz continuous with $\overline{g}$ the smoothed model and Φ the standard normal cumulative distribution function.
2. We prove a certified radius of robustness for input-dependent randomized smoothing based on our Lipschitz bound, which, unlike prior works, enjoys an increase without bound as the model confidence increases, does not rely on memory-based methods, and rigorously permits the Gaussian smoothing noise standard deviation mapping to be learned via Lipschitz continuous neural networks with general structure. Our Lipschitz bound and certified radius reduce exactly to those of standard RS as a special case.
3. We conduct proof-of-concept numerical experiments on MNIST that demonstrates "best-of-both-worlds" performance; our input-dependent smoothing approach simultaneously matches the high clean accuracy of low-variance standard RS *and* the improved robustness of high-variance standard RS, with a negligible increase in certification time.

To streamline presentation, proofs are deferred to Appendix A.

2 Preliminaries

2.1 Notations

The $d \times d$ identity matrix is denoted by I_d. The indicator function on a set A is defined by $\mathbf{1}_A(x) = 1$ if $x \in A$ and $\mathbf{1}_A(x) = 0$ if $x \notin A$. The multivariate normal distribution with mean $\mu \in \mathbb{R}^d$ and

covariance matrix $\Sigma \in \mathbb{S}_+^d$ is denoted by $\mathcal{N}(\mu, \Sigma)$, where $\mathbb{S}_+^d$ denotes the set of $d \times d$ positive semidefinite matrices. The probability density function (PDF) and cumulative distribution function (CDF) of the standard normal distribution $\mathcal{N}(0, 1)$ are denoted by ϕ and Φ, respectively. We will also occasionally write $\phi_{\mu,\Sigma}$ to denote the PDF of the multivariate normal distribution $\mathcal{N}(\mu, \Sigma)$. The PDF and CDF of the χ_d^2 distribution (with d degrees of freedom) are denoted by $f_{\chi_d^2}$ and $F_{\chi_d^2}$, respectively. Recall that a function $f\colon \mathbb{R}^d \to \mathbb{R}$ is said to be L-Lipschitz continuous (or L-Lipschitz for short) if $|f(x) - f(x')| \leq L\|x - x'\|_2$ for some associated $L \geq 0$, where $\|\cdot\|_2$ denotes the ℓ_2-norm. The Lipschitz constant $\mathrm{Lip}(f)$ is used to denote any valid $L \geq 0$ such that f is L-Lipschitz. If M is a matrix, then $\|M\|$ denotes the spectral norm of M. We use $\mathrm{diag}(a_1, \ldots, a_d)$ to denote the $d \times d$ diagonal matrix with a_i being the ith diagonal element for all $i \in \{1, \ldots, d\}$. The rectified linear unit is defined by $\mathrm{ReLU}\colon x \mapsto \max\{0, x\}$, applied elementwise to vectors x.

2.2 Problem Statement

Consider a pre-trained n-class classifier $f\colon \mathbb{R}^d \to \{1, \ldots, n\}$ given by

$$f(x) \in \operatorname*{arg\,max}_{i \in \{1,\ldots,n\}} g_i(x), \tag{1}$$

with $g\colon \mathbb{R}^d \to \mathbb{R}^n$. Standard, input-independent RS robustifies the classifier by choosing a *fixed* standard deviation $\sigma > 0$ and replacing the prediction $f(x)$ by

$$\overline{f}(x) \in \operatorname*{arg\,max}_{i \in \{1,\ldots,n\}} \overline{g}(x),$$

with $\overline{g}\colon \mathbb{R}^d \to \mathbb{R}^n$ being a smoothed version of g:

$$\overline{g}(x) = \mathbb{E}_{\epsilon \sim \mathcal{N}(0, \sigma^2 I_d)}[g(x + \epsilon)].$$

This formulation of RS, which averages the model's output scores, is sometimes called "soft smoothing" [17], and is known to converge to "hard smoothing," as originally formulated in Cohen et al. [8], in the case that $g(x) \in [0, 1]^n$ is a post-softmax probability vector and the pre-softmax temperature scaling increases (making $g(x)$ approach a standard unit vector). We stick with the soft smoothing formulation as it is more commonly used in input-dependent smoothing schemes and is more general than hard smoothing.

Input-dependent randomized smoothing generalizes standard RS by allowing the Gaussian noise parameters to vary with the input, i.e., $\overline{g}$ takes the more general form

$$\overline{g}(x) \coloneqq \mathbb{E}_{\epsilon \sim \mathcal{N}(\mu(x), \Sigma(x))}[g(x + \epsilon)], \tag{2}$$

with $\mu\colon \mathbb{R}^d \to \mathbb{R}^d$ and $\Sigma\colon \mathbb{R}^d \to \mathbb{S}_+^d$ being mean and covariance matrix mappings.

In both standard and input-dependent RS, the noising process intuitively drowns out the effects of any possible manipulations of the input data. Mathematically, this amounts to smoothing the decision boundaries of the model, as (standard) RS can be viewed as convolution of the base classifier with a (fixed) Gaussian mollifier. Indeed, this idea is well known to culminate into provable robustness guarantees for the smoothed classifier $\overline{f}$ in the case of standard RS:

Theorem 1 (Cohen et al. [8], Zhai et al. [17])**.** *Assume that $g(x) \in [0, 1]^n$ for all $x \in \mathbb{R}^d$. Let $\sigma > 0$. Consider a point $x \in \mathbb{R}^d$, let $y = \overline{f}(x)$ be the classification of x under the standard RS classifier $\overline{f}$ with fixed smoothing distribution $\mathcal{N}(0, \sigma^2 I_d)$, and let $y' \in \arg\max_{i \in \{1,\ldots,n\} \setminus \{y\}} \overline{g}_i(x)$ be a corresponding runner-up class. Then, it holds that*

$$\overline{f}(x + \delta) = y$$

for all $\delta \in \mathbb{R}^d$ such that

$$\|\delta\|_2 \leq r(x, y) \coloneqq \frac{\sigma}{2}\left(\Phi^{-1}(\overline{g}_y(x)) - \Phi^{-1}(\overline{g}_{y'}(x))\right). \tag{3}$$

The value $r(x, y) \geq 0$ in (3) is called the *certified radius* of the model at the input-label pair (x, y). A key feature of the certified radius formula in (3) is the presence of Φ^{-1}, which allows for the radius to enjoy an unbounded increase as the model confidence $\overline{g}_y(x)$ increases (resulting in strong

robustness guarantees). This functional form is very delicate to achieve, as it amounts to proving the Lipschitz continuity of every $\Phi^{-1} \circ \overline{g}_i$, despite neither Φ^{-1} nor $\overline{g}_i$ being Lipschitz. As such, the added complexity of input-dependent smoothing schemes makes proving certified radii with the same, strong functional form of (3) mathematically challenging (and it is certainly not as trivial as replacing σ with $\sigma(x)$ in (3), as this is *not* a valid certified radius in the input-dependent case [13]). Consequently, input-dependent certified radii, up until this point, have either taken much weaker, uniformly bounded forms without the Φ^{-1} [13, 27], or have been restricted to use memory-based methods with piecewise-constant standard deviations [23, 22, 24]. The overarching goal of our work is to overcome these limitations by proving an input-dependent certified radius taking the strong functional form of (3), without relying on memory-based methods, and while allowing for the standard deviation mapping $x \mapsto \sigma(x)$ to be implemented as a neural network and optimized from data. We now move to our novel theoretical analysis to achieve this goal.

3 Theoretical Analysis and Certified Radius of Input-Dependent Smoothing

Throughout the remainder of the paper, we consider input-dependent RS taking the form (2), unless otherwise specified. To prove a certified radius for the model, we begin by analyzing the Lipschitz continuity of $\Phi^{-1} \circ \overline{g}_i$. Although this function is known to be $\frac{1}{\sigma}$-Lipschitz when using the standard RS smoothing distribution $\mathcal{N}(0, \sigma^2 I_d)$ with fixed $\sigma > 0$ [17, Lemma 1], it is not evident how the Lipschitz continuity is affected by the use of input-dependent smoothing distributions. We will assume throughout that $x \mapsto \mu(x)$ and $x \mapsto \Sigma(x)$ are both differentiable maps. In general, however, one may still suspect that the model will become poorly behaved in cases where the distribution $\mathcal{N}(\mu(x), \Sigma(x))$ varies wildly with x (which is entirely possible; differentiability of the parameters μ and Σ does not imply Lipschitz continuity of them). We begin establishing our guarantees for the strong Lipschitz property of the model by stating the following simplifying assumption.

Assumption 1. For all $x \in \mathbb{R}^d$, it holds that $\Sigma(x)$ is diagonal;

$$\Sigma(x) = \operatorname{diag}(\sigma_1^2(x), \ldots, \sigma_d^2(x))$$

for some $\sigma_1(x), \ldots, \sigma_d(x) > 0$.

The following three technical lemmas will be key in establishing Lipschitzness of $\Phi^{-1} \circ \overline{g}_i$ in our input-dependent setting:

Lemma 1. *Assume that Assumption 1 holds. Let $x, \epsilon \in \mathbb{R}^d$. It holds that*

$$\nabla_x \phi_{\mu(x),\Sigma(x)}(\epsilon - x) = \phi_{\mu(x),\Sigma(x)}(\epsilon - x) \Bigg((I_d + D\mu(x))\Sigma^{-1}(x)(\epsilon - x - \mu(x)) + \sum_{i=1}^{d} \frac{(\epsilon_i - x_i - \mu_i(x))^2 - \sigma_i^2(x)}{\sigma_i^3(x)} \nabla \sigma_i(x) \Bigg),$$

where

$$D\mu(x) = \begin{bmatrix} \nabla \mu_1(x) & \cdots & \nabla \mu_d(x) \end{bmatrix}.$$

Intuitively, Lemma 1 allows us to characterize how much of the variation in the final model $\overline{g}$ is due to the inherent variation in the shape of the PDF $\phi_{\mu(x),\Sigma(x)}(\cdot)$ (the term in the gradient with I_d), the variation of μ (the term with $D\mu(x)$), and the variation of Σ (the term with $\nabla \sigma_i(x)$). All three forms of variation contribute to the Lipschitzian quality (or lack thereof) of $\overline{g}$.

Lemma 2. *Let $w \in \mathbb{R}^d$, let $p \in [0, 1]$, let $l \colon \mathbb{R}^d \to \mathbb{R}$, and let ν be a probability measure on $\mathbb{R}^d$ (that possibly depends on w). If there exists $t_p \in \mathbb{R}$ such that $\mathbb{P}_{\epsilon \sim \nu}(l(\epsilon) \geq t_p) = p$, then $h \colon y \mapsto \mathbf{1}_{\{z \in \mathbb{R}^d : l(z) \geq t_p\}}(y - w)$ solves the functional optimization problem*

$$\sup \left\{ \mathbb{E}_{\epsilon \sim \nu}[l(\epsilon) f(w + \epsilon)] : \mathbb{E}_{\epsilon \sim \nu}[f(w + \epsilon)] = p,\ f \colon \mathbb{R}^d \to [0, 1] \right\}.$$

Lemma 2 takes inspiration from the approach taken in Salman et al. [16, Lemma 2], although our result is proven for more general weighting functions l and probability measures ν, which is important for proving robustness in our input-dependent smoothing setting.

Lemma 3. *It holds that* $R\colon (0,1) \to (0,\infty)$ *defined by*

$$R(p) = \frac{\epsilon_0(p) f_{\chi^2_d}(\epsilon_0(p))}{\phi(\Phi^{-1}(p))}$$

is monotone decreasing on $[p_{\min}, 1-p_{\min}]$ *for some* $p_{\min} \in (0, 1/2)$*, where* $\epsilon_0\colon (0,1) \to (0,\infty)$ *is defined by*

$$\epsilon_0(p) = F^{-1}_{\chi^2_d}(1-p). \tag{4}$$

The ratio $R(p)$ in Lemma 3 appears naturally as a key factor in our Lipschitz bound (to come in Theorem 2), and it is beneficial to be able to verify that this ratio is monotone decreasing in p on as large of an interval $[p_{\min}, 1-p_{\min}]$ as possible (clearly, this makes the theoretical result in Lemma 3 stronger). In practice, the minimum probability value $p_{\min}$ in Lemma 3 can be taken to be *very* close to zero for realistic dimensions d, which can be seen by plotting the function R. For instance, when $d = 784$, graphical methods show that R is monotone decreasing on $[p_{\min}, 1-p_{\min}]$ with $p_{\min} = 10^{-10}$. Analytically proving a (tight) upper bound on the minimal valid value of $p_{\min}$ that ensures monotonicity of R poses an interesting open problem.

Utilizing the above three key technical lemmas, we are now able to establish the strong Lipschitz property for input-dependent randomized smoothing, in the case that μ is identically zero, and $\Sigma(x) = \sigma^2(x) I_d$:

Theorem 2. *Consider zero-mean isotropic input-dependent randomized smoothing with* $\mu(x) = 0$ *and* $\Sigma(x) = \sigma^2(x) I_d$ *for all* $x \in \mathbb{R}^d$*. Let* $i \in \{1, \ldots, n\}$ *be arbitrary. If* σ *is Lipschitz continuous and bounded below by* $\sigma_{\min} > 0$*, and if* $\overline{g}_i(x) \in [p_{\min}, 1-p_{\min}]$ *for* $p_{\min} \in (0, 1/2)$ *as in Lemma 3, then* $x \mapsto \Phi^{-1}(\overline{g}_i(x))$ *is* L*-Lipschitz with*

$$L = \frac{1}{\sigma_{\min}} \left(1 + 2\,\mathrm{Lip}(\sigma) \frac{\epsilon_0(p_{\min}) f_{\chi^2_d}(\epsilon_0(p_{\min}))}{\phi(\Phi^{-1}(p_{\min}))} \right), \tag{5}$$

where $\epsilon_0\colon (0,1) \to (0,\infty)$ *is defined as in* (4)*.*

Remark 1. The Lipschitz constant (5) recovers that of standard, input-*independent* randomized smoothing, namely $L = \frac{1}{\sigma}$, in the case that σ is taken to be a constant (since $\mathrm{Lip}(\sigma) = 0$ in this case).

Remark 2. It is easy to ensure that all hypotheses in Theorem 2 are satisfied in practice. Specifically, $\sigma\colon x \mapsto \sigma(x)$ can be constructed to be a single-output real-valued neural network with final layer taking the form $z \mapsto \mathrm{ReLU}(z) + \sigma_{\min}$, with $\sigma_{\min} > 0$ being a fixed hyperparameter. This network is easily constrained to be Lipschitz continuous with any desired Lipschitz constant $\mathrm{Lip}(\sigma)$ by using standard weight normalization methods. Finally, it can be ensured that $\overline{g}_i(x) \in [p_{\min}, 1-p_{\min}]$ for every class $i \in \{1, \ldots, n\}$ by normalizing the outputs of the base classifier according to $g(x) = (1 - n p_{\min}) g_{[0,1]}(x) + p_{\min} 1_n$, where $g_{[0,1]}(x)$ is the (standard output) vector of scores with elements in $[0,1]$, typically computed using a softmax, and where 1_n denotes the n-vector of all ones.

Theorem 2 gives rise to a certified radius for input-dependent smoothing with zero-mean isotropic Gaussian smoothing distributions:

Theorem 3. *Consider input-dependent smoothing, with all hypotheses of Theorem 2 satisfied;* $\mu(x) = 0$*,* $\Sigma(x) = \sigma^2(x) I_d$ *with* σ *Lipschitz continuous and bounded below by* $\sigma_{\min} > 0$*, and* $\overline{g}(x) \in [p_{\min}, 1-p_{\min}]^n$ *for* $p_{\min} \in (0, 1/2)$ *as in Lemma 3. Consider a point* $x \in \mathbb{R}^d$*, let* $y = \overline{f}(x)$ *be the classification of* x *under the smoothed classifier, and let* $y' \in \arg\max_{i \in \{1,\ldots,n\} \setminus \{y\}} \overline{g}_i(x)$ *be the runner-up class. Then, it holds that*

$$\overline{f}(x + \delta) = y$$

for all $\delta \in \mathbb{R}^d$ *such that*

$$\|\delta\|_2 \leq \overline{r}(x, y) \coloneqq \frac{1}{2L} \left(\Phi^{-1}(\overline{g}_y(x)) - \Phi^{-1}(\overline{g}_{y'}(x)) \right),$$

with L *being the Lipschitz constant defined in* (5)*.*

Theorem 3 achieves our overarching goal in the case with $\mu(x) = 0$ and $\Sigma(x) = \sigma^2(x) I_d$. Specifically, we rigorously obtain a certified radius for input-dependent smoothing taking the strong functional form of (3) with Φ^{-1}, while allowing for $x \mapsto \sigma(x)$ to be a general (Lipschitz and uniformly lower-bounded) neural network learned from data, and while avoiding memory-based methods with piecewise-constant σ.

4 Numerical Experiments

We conduct numerical experiments on the MNIST dataset of handwritten digits [30]. The experiments are run on a commercially available laptop computer using PyTorch [31] and SciPy [32]. We use a two-layer fully-connected base classifier g with hidden layer width 200 and ReLU activation functions. The base classifier is trained using PyTorch's default optimizer together with 40-step ℓ_2-projected gradient descent (PGD) adversarial training [4], with attack magnitudes $\|\delta\|_2 = 1$ and step size 0.01. Such ℓ_2-adversarial training approaches are known to yield state-of-the-art certified radii under standard Gaussian RS, even compared to training using Gaussian augmentation [16, 33]. The base classifier is trained to 98% accuracy on clean test data. The base classifier output probabilities are normalized according to the method described in Remark 2, with $p_{\text{min}} = 10^{-7}$.

We compare our input-dependent randomized smoothing (IDRS) method to standard RS [8] using a variety of (input-*independent*) Gaussians. Specifically, we include the common baseline standard deviations of $\sigma \in \{0.25, 0.5, 1\}$, used in Cohen et al. [8], Salman et al. [16], Jeong et al. [33]. Our input-dependent model is implemented using the zero-mean, isotropic Gaussian smoothing scheme $\mathcal{N}(0, \sigma_\theta^2(x) I_d)$ by constructing $\sigma_\theta \colon \mathbb{R}^d \to [\sigma_{\text{min}}, \infty)$ as a two-layer neural network, parameterized by $\theta \in \mathbb{R}^p$ consisting of all weights and biases, with ReLU activation functions, and hidden layer width 200. We utilize a minimum standard deviation value of $\sigma_{\text{min}} = 0.65$. We explicitly constrain the Lipschitz constant of this network to $\mathrm{Lip}(\sigma) = 0.5$, using spectral weight normalization at every step of the learning process [34]. The neural network σ_θ is learned to maximize the average certified radius, as calculated by Theorem 3, by solving

$$\sup_{\theta \in \mathbb{R}^p} \left(-\frac{1}{N} \sum_{i=1}^{N} \overline{r}(x_i, y_i) + \lambda \sum_{j=1}^{n_W} \|W_j^\theta\| \right),$$

where $(x_1, y_1), \ldots, (x_N, y_N)$ is a collection of labeled training data, $W_1^\theta, \ldots, W_{n_W}^\theta$ denote the n_W weight matrices collected in θ, and, of course, the radii $\overline{r}(x_i, y_i)$ depend implicitly on θ. We use spectral regularization parameter $\lambda = 0.01$, which we remark is performed in addition to the spectral weight normalization.

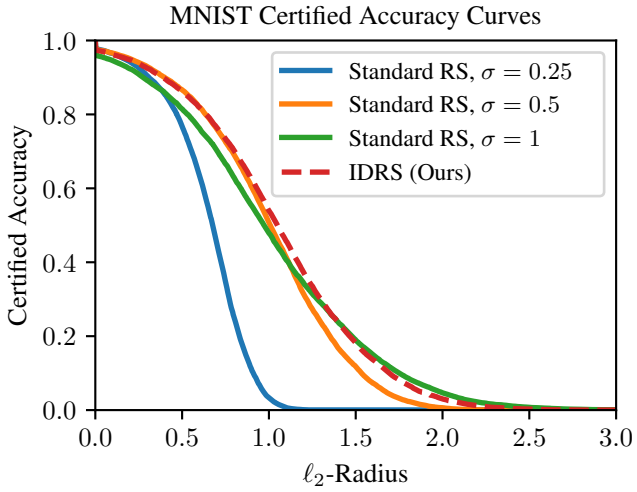

Figure 1: Certified accuracy curves for our input-dependent randomized smoothing method (IDRS), as well as three standard RS curves corresponding to various standard deviations σ. Our method is seen to enjoy a "best-of-both-worlds" when compared to the moderate $\sigma = 0.5$ and large $\sigma = 1$ standard RS curves.

The certified accuracy curves across a range of attack radii, computed using $n_{\text{samples}} = 1000$ noise samples per input, are plotted in Figure 1 for each method. We see that our method (IDRS) exhibits "best-of-both-worlds" performance relative to standard smoothing with moderate $\sigma = 0.5$ and large $\sigma = 1$. Specifically, our model matches the high accuracy of low-strength standard smoothing (both

$\sigma = 0.25$ and $\sigma = 0.5$) at small attack radii, in addition to matching the high robustness of high-strength standard smoothing ($\sigma = 1$) at large attack radii. Contrarily, the standard RS curves exhibit a strict weakness in performance compared to IDRS, either on clean data (when the standard smoothing is too strong), or on strongly attacked data (when the standard smoothing is too weak). Since our smoothing scheme only requires one additional forward pass through the scalar-valued network σ_θ when compared to standard RS, we observe a negligible increase in the average certification time required for each test input; see Table 1.

Table 1: Average certification times per test input.

Method	Time (milliseconds)
IDRS (Ours)	9.286
Standard RS, $\sigma = 0.25$	9.225
Standard RS, $\sigma = 0.5$	9.244
Standard RS, $\sigma = 1$	9.271

5 Conclusions and future work

In this paper, we consider input-dependent randomized smoothing as a means to increase certified robustness by allowing the smoothing distribution to learn how to adjust its shape according to the data distribution. In the zero-mean, isotropic Gaussian case, we prove a strong Lipschitz property of the input-dependent RS model, under suitable assumptions on the standard deviation mapping that are easily enforced in practice. We use our Lipschitz property to derive a certified radius of robustness for the input-dependent RS model, which, to the best of our knowledge, is the first non-memory-based radius to outperform standard RS. Our Lipschitz constant and certified radius recover those of standard, input-*independent* smoothing as a special case.

Future theoretical work includes extending our robustness guarantees to nonzero means and anisotropic variances, certifying anisotropic regions of the input space rather than (isotropic) ℓ_2-balls, analytically upper-bounding the minimal p_{min} ensuring validity of our theory, and characterizing the behavior of $R(p_{\text{min}})$—a key factor in our certified radius—as the dimensionality d increases. Experimentally, we seek to extend our implementations to larger-scale benchmark problems such as the CIFAR-10 and ImageNet datasets, and to compare against additional baselines.

References

[1] 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.

[2] 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.

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

[4] 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.

[5] Brendon G. Anderson, Ziye Ma, Jingqi Li, and Somayeh Sojoudi. Towards optimal branching of linear and semidefinite relaxations for neural network robustness certification. *Journal of Machine Learning Research*, 26(81):1–59, 2025.

[6] Shiqi Wang, Huan Zhang, Kaidi Xu, Xue Lin, Suman Jana, Cho-Jui Hsieh, and Zico Kolter. Beta-CROWN: Efficient bound propagation with per-neuron split constraints for neural network robustness verification. In *Advances in Neural Information Processing Systems*, volume 34, pages 29909–29921, 2021.

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

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

[9] 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, 2019.

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

[11] Yatong Bai, Brendon G. Anderson, Aerin Kim, and Somayeh Sojoudi. Improving the accuracy-robustness trade-off of classifiers via adaptive smoothing. *SIAM Journal on Mathematics of Data Science*, 6(3):788–814, 2024.

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

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

[14] Jeet Mohapatra, Ching-Yun Ko, Lily Weng, Pin-Yu Chen, Sijia Liu, and Luca Daniel. Hidden cost of randomized smoothing. In *International Conference on Artificial Intelligence and Statistics*, pages 4033–4041. PMLR, 2021.

[15] 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.

[16] 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.

[17] 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.

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

[19] Song Xia, Yi Yu, Xudong Jiang, and Henghui Ding. Mitigating the curse of dimensionality for certified robustness via dual randomized smoothing. In *International Conference on Learning Representations*, 2024.

[20] Bo-Han Kung and Shang-Tse Chen. Towards large certified radius in randomized smoothing using quasiconcave optimization. In *AAAI Conference on Artificial Intelligence*, 2024.

[21] 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`.

[22] Francisco Eiras, Motasem Alfarra, M. Pawan Kumar, Philip H.S. Torr, Puneet K. Dokania, Bernard Ghanem, and Adel Bibi. ANCER: Anisotropic certification via sample-wise volume maximization. *Transactions on Machine Learning Research*, 2022.

[23] Motasem Alfarra, Adel Bibi, Philip H.S. Torr, and Bernard Ghanem. Data dependent randomized smoothing. In *Uncertainty in Artificial Intelligence*, pages 64–74, 2022.

[24] Taras Rumezhak, Francisco Girbal Eiras, Philip H.S. Torr, and Adel Bibi. RANCER: Non-axis aligned anisotropic certification with randomized smoothing. In *IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 4672–4680, 2023.

[25] Peter Súkeník, Aleksei Kuvshinov, and Stephan Günnemann. Intriguing properties of input-dependent randomized smoothing. In *International Conference on Machine Learning*, pages 20697–20743. PMLR, 2022.

[26] Chen Chen, Kezhi Kong, Peihong Yu, Juan Luque, Tom Goldstein, and Furong Huang. Insta-RS: Instance-wise randomized smoothing for improved robustness and accuracy. *arXiv preprint arXiv:2103.04436*, 2021.

[27] Brendon G. Anderson, Samuel Pfrommer, and Somayeh Sojoudi. Towards optimal randomized smoothing: A semi-infinite linear programming approach. In *ICML Workshop on Formal Verification of Machine Learning (WFVML)*, 2022.

[28] Saiyue Lyu, Shadab Shaikh, Frederick Shpilevskiy, Evan Shelhamer, and Mathias Lécuyer. Adaptive randomized smoothing: Certified adversarial robustness for multi-step defences. In *Advances in Neural Information Processing Systems*, volume 37, pages 134043–134074, 2024.

[29] Ruoxin Chen, Jie Li, Junchi Yan, Ping Li, and Bin Sheng. Input-specific robustness certification for randomized smoothing. In *AAAI Conference on Artificial Intelligence*, pages 6295–6303, 2022.

[30] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 2002.

[31] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural information Processing Systems*, 2019.

[32] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, et al. SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nature Methods*, 17(3): 261–272, 2020.

[33] Jongheon Jeong, Sejun Park, Minkyu Kim, Heung-Chang Lee, Do-Guk Kim, and Jinwoo Shin. SmoothMix: Training confidence-calibrated smoothed classifiers for certified robustness. In *Advances in Neural Information Processing Systems*, volume 34, pages 30153–30168, 2021.

[34] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *International Conference on Learning Representations*, 2018.

[35] Richard F. Lyon. On closed-form tight bounds and approximations for the median of a gamma distribution. *PLoS One*, 16(5), 2021.

A Proofs

Lemma 1. *Assume that Assumption 1 holds. Let $x, \epsilon \in \mathbb{R}^d$. It holds that*

$$\nabla_x \phi_{\mu(x),\Sigma(x)}(\epsilon - x) = \phi_{\mu(x),\Sigma(x)}(\epsilon - x)\Bigg((I_d + D\mu(x))\Sigma^{-1}(x)(\epsilon - x - \mu(x)) + \sum_{i=1}^{d} \frac{(\epsilon_i - x_i - \mu_i(x))^2 - \sigma_i^2(x)}{\sigma_i^3(x)} \nabla\sigma_i(x)\Bigg),$$

where

$$D\mu(x) = \left[\nabla\mu_1(x) \quad \cdots \quad \nabla\mu_d(x)\right].$$

Proof of Lemma 1. This follows from a routine application of chain and product rules when computing

$$\nabla_x \phi_{\mu(x),\Sigma(x)}(\epsilon - x) = \nabla_x \left(\frac{1}{(2\pi)^{d/2} \prod_{i=1}^d \sigma_i(x)} \exp\left(-\frac{1}{2} \sum_{i=1}^{d} \frac{(\epsilon_i - x_i - \mu_i(x))^2}{\sigma_i^2(x)} \right) \right).$$

□

Lemma 2. *Let $w \in \mathbb{R}^d$, let $p \in [0,1]$, let $l\colon \mathbb{R}^d \to \mathbb{R}$, and let ν be a probability measure on $\mathbb{R}^d$ (that possibly depends on w). If there exists $t_p \in \mathbb{R}$ such that $\mathbb{P}_{\epsilon\sim\nu}(l(\epsilon) \geq t_p) = p$, then $h\colon y \mapsto \mathbf{1}_{\{z\in\mathbb{R}^d : l(z) \geq t_p\}}(y - w)$ solves the functional optimization problem*

$$\sup\left\{ \mathbb{E}_{\epsilon\sim\nu}[l(\epsilon) f(w+\epsilon)] : \mathbb{E}_{\epsilon\sim\nu}[f(w+\epsilon)] = p,\ f\colon \mathbb{R}^d \to [0,1] \right\}.$$

Proof of Lemma 2. First, notice that $h(y) = \mathbf{1}_{\{z\in\mathbb{R}^d : l(z)\geq t_p\}}(y-w) \in \{0,1\} \subseteq [0,1]$ for all $y \in \mathbb{R}^d$, and

$$\mathbb{E}_{\epsilon\sim\nu}[h(w+\epsilon)] = \mathbb{E}_{\epsilon\sim\nu}[\mathbf{1}_{\{z\in\mathbb{R}^d : l(z)\geq t_p\}}(\epsilon)] = \mathbb{P}_{\epsilon\sim\nu}(l(\epsilon)\geq t_p) = p,$$

so h is feasible. To show that h is optimal, let f be any other feasible function. Then, we have that

$$\begin{aligned}
\mathbb{E}_{\epsilon\sim\nu}[l(\epsilon)h(w+\epsilon)] - \mathbb{E}_{\epsilon\sim\nu}[l(\epsilon)f(w+\epsilon)] &= \mathbb{E}_{\epsilon\sim\nu}[l(\epsilon)(h(w+\epsilon) - f(w+\epsilon))] \\
&= \mathbb{E}_{\epsilon\sim\nu}[l(\epsilon)(h(w+\epsilon) - f(w+\epsilon))] \\
&\quad - t_p\mathbb{E}_{\epsilon\sim\nu}[h(w+\epsilon) - f(w+\epsilon)] \\
&= \mathbb{E}_{\epsilon\sim\nu}[(l(\epsilon) - t_p)(h(w+\epsilon) - f(w+\epsilon))],
\end{aligned}$$

where the second equality comes from the fact that $\mathbb{E}_{\epsilon\sim\nu}[f(w+\epsilon)] = p = \mathbb{E}_{\epsilon\sim\nu}[h(w+\epsilon)]$. Splitting the integral, it follows that

$$\begin{aligned}
&\mathbb{E}_{\epsilon\sim\nu}[l(\epsilon)h(w+\epsilon)] - \mathbb{E}_{\epsilon\sim\nu}[l(\epsilon)f(w+\epsilon)] \\
&\quad= \int_{\epsilon\in\{z\in\mathbb{R}^d : l(z)\geq t_p\}} (l(\epsilon) - t_p)(h(w+\epsilon) - f(w+\epsilon))d\nu(\epsilon) \\
&\qquad+ \int_{\epsilon\in\{z\in\mathbb{R}^d : l(z)< t_p\}} (l(\epsilon) - t_p)(h(w+\epsilon) - f(w+\epsilon))d\nu(\epsilon).
\end{aligned}$$

The first above integral is nonnegative, since, for all $\epsilon \in \{z \in \mathbb{R}^d : l(z) \geq t_p\}$, it holds that $l(\epsilon) \geq t_p$ and $h(w+\epsilon) - f(w+\epsilon) = 1 - f(w+\epsilon) \geq 0$, implying that the integrand is nonnegative. The second above integral is also nonnegative, since, for all $\epsilon \in \{z \in \mathbb{R}^d : l(z) < t_p\}$, it holds that $l(\epsilon) < t_p$ and $h(w+\epsilon) - f(w+\epsilon) = 0 - f(w+\epsilon) \leq 0$, implying that the integrand is nonnegative. Thus, it holds that

$$\mathbb{E}_{\epsilon\sim\nu}[l(\epsilon)h(w+\epsilon)] - \mathbb{E}_{\epsilon\sim\nu}[l(\epsilon)f(w+\epsilon)] \geq 0.$$

Since f was chosen to be an arbitrary feasible point for the optimization, this proves that h is optimal. □

Lemma 3. *It holds that* $R\colon (0,1) \to (0,\infty)$ *defined by*

$$R(p) = \frac{\epsilon_0(p) f_{\chi^2_d}(\epsilon_0(p))}{\phi(\Phi^{-1}(p))}$$

is monotone decreasing on $[p_{\min}, 1-p_{\min}]$ *for some* $p_{\min} \in (0, 1/2)$*, where* $\epsilon_0\colon (0,1) \to (0,\infty)$ *is defined by*

$$\epsilon_0(p) = F^{-1}_{\chi^2_d}(1-p). \tag{4}$$

Proof of Lemma 3. It suffices to prove that $p \mapsto \log R(p)$ is monotone decreasing on $[p_{\min}, 1-p_{\min}]$ for some $p_{\min} \in (0, 1/2)$. We show this by computing the derivative of $p \mapsto \log R(p)$.

First, notice that $F_{\chi^2_d}(\epsilon_0(p)) = 1-p$, and therefore $f_{\chi^2_d}(\epsilon_0(p))\epsilon_0'(p) = -1$, implying that

$$\epsilon_0'(p) = -\frac{1}{f_{\chi^2_d}(\epsilon_0(p))}.$$

Furthermore, the function $u\colon p \mapsto \Phi^{-1}(p)$ satisfies $\Phi(u(p)) = p$, implying that $\phi(u(p))u'(p) = 1$, and therefore

$$\frac{d}{dp}\Phi^{-1}(p) = u'(p) = \frac{1}{\phi(u(p))} = \frac{1}{\phi(\Phi^{-1}(p))}.$$

Thus,

$$\begin{aligned}
\frac{d}{dp}\log R(p) &= \frac{d}{dp}\left(\log\epsilon_0(p) + \log f_{\chi^2_d}(\epsilon_0(p)) - \log\phi(\Phi^{-1}(p))\right) \\
&= \frac{\epsilon_0'(p)}{\epsilon_0(p)} + \frac{f'_{\chi^2_d}(\epsilon_0(p))\epsilon_0'(p)}{f_{\chi^2_d}(\epsilon_0(p))} - \frac{\phi'(\Phi^{-1}(p))\frac{d}{dp}\Phi^{-1}(p)}{\phi(\Phi^{-1}(p))} \\
&= -\frac{1}{f_{\chi^2_d}(\epsilon_0(p))}\left(\frac{1}{\epsilon_0(p)} + \frac{f'_{\chi^2_d}(\epsilon_0(p))}{f_{\chi^2_d}(\epsilon_0(p))}\right) - \frac{\phi'(\Phi^{-1}(p))\frac{1}{\phi(\Phi^{-1}(p))}}{\phi(\Phi^{-1}(p))}.
\end{aligned}$$

It is easy to show the following derivatives for the standard normal and chi-squared probability density functions:

$$\phi'(u) = -u\phi(u)$$
$$f'_{\chi^2_d}(v) = \left(\frac{d/2-1}{v} - \frac{1}{2}\right) f_{\chi^2_d}(v).$$

Substituting these expressions into our derivative of interest gives

$$\begin{aligned}\frac{d}{dp}\log R(p) &= -\frac{1}{f_{\chi^2_d}(\epsilon_0(p))}\left(\frac{1}{\epsilon_0(p)} + \frac{d/2-1}{\epsilon_0(p)} - \frac{1}{2}\right) + \frac{\Phi^{-1}(p)}{\phi(\Phi^{-1}(p))} \\ &= -\frac{1}{f_{\chi^2_d}(\epsilon_0(p))}\left(\frac{d/2}{\epsilon_0(p)} - \frac{1}{2}\right) + \frac{\Phi^{-1}(p)}{\phi(\Phi^{-1}(p))}.\end{aligned}$$

At $p = 1/2$, it holds that $\Phi^{-1}(p) = 0$, and that

$$\epsilon_0(p) = F^{-1}_{\chi^2_d}(1/2),$$

which equals the median of the chi-squared distribution χ^2_d. Since this distribution is an instance of the gamma distribution with shape parameter $d/2$ and scale parameter 2, Lyon [35] gives that the median is bounded as

$$F^{-1}_{\chi^2_d}(1/2) < d.$$

Therefore, we find that

$$\frac{d}{dp}\log R(p)\bigg|_{p=1/2} < -\frac{1}{f_{\chi^2_d}(\epsilon_0(p))}\left(\frac{d/2}{d} - \frac{1}{2}\right) = 0.$$

Clearly, $p \mapsto \frac{d}{dp}\log R(p)$ is continuous on $(0,1)$, and therefore this shows that there exists $\delta \in (0, 1/2)$ such that

$$\frac{d}{dp}\log R(p) < 0 \text{ for all } p \in [1/2-\delta, 1/2+\delta].$$

Thus, $p \mapsto \log R(p)$ is monotone decreasing on $[p_{\min}, 1-p_{\min}]$, where $p_{\min} = 1/2-\delta \in (0, 1/2)$. □

Theorem 2. *Consider zero-mean isotropic input-dependent randomized smoothing with $\mu(x) = 0$ and $\Sigma(x) = \sigma^2(x)I_d$ for all $x \in \mathbb{R}^d$. Let $i \in \{1, \ldots, n\}$ be arbitrary. If σ is Lipschitz continuous and bounded below by $\sigma_{\min} > 0$, and if $\overline{g}_i(x) \in [p_{\min}, 1-p_{\min}]$ for $p_{\min} \in (0, 1/2)$ as in Lemma 3, then $x \mapsto \Phi^{-1}(\overline{g}_i(x))$ is L-Lipschitz with*

$$L = \frac{1}{\sigma_{\min}}\left(1 + 2\,\mathrm{Lip}(\sigma)\frac{\epsilon_0(p_{\min}) f_{\chi^2_d}(\epsilon_0(p_{\min}))}{\phi(\Phi^{-1}(p_{\min}))}\right), \tag{5}$$

where $\epsilon_0 \colon (0,1) \to (0,\infty)$ is defined as in (4)*.*

Proof of Theorem 2. To simplify exposition, we drop the model output's subscript notation and write g and $\overline{g}$ in place of g_i and $\overline{g}_i$, respectively. The desired Lipschitz constant of $x \mapsto \Phi^{-1}(\overline{g}(x))$ can be obtained by bounding the norm of the gradient

$$\nabla_x \Phi^{-1}(\overline{g}(x)) = \frac{\nabla \overline{g}(x)}{\Phi'(\Phi^{-1}(\overline{g}(x)))}. \tag{6}$$

Notice that the denominator is readily computed as

$$\Phi'(\Phi^{-1}(\overline{g}(x))) = \phi(\Phi^{-1}(\overline{g}(x))) = \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}(\Phi^{-1}(\overline{g}(x)))^2\right),$$

where $\phi \colon \mathbb{R} \to \mathbb{R}$ denotes the univariate density function of the standard normal distribution $\mathcal{N}(0,1)$. Thus, we precisely know the denominator in the gradient (6) of interest, so everything boils down to computing the numerator, i.e., the gradient $\nabla \overline{g}(x)$.

We have that

$$\begin{aligned}\nabla \overline{g}(x) &= \nabla_x \int_{\mathbb{R}^d} g(x+\epsilon)\phi_{\mu(x),\Sigma(x)}(\epsilon)d\epsilon \\ &= \nabla_x \int_{\mathbb{R}^d} g(\epsilon)\phi_{\mu(x),\Sigma(x)}(\epsilon - x)d\epsilon \\ &= \int_{\mathbb{R}^d} g(\epsilon)\nabla_x \phi_{\mu(x),\Sigma(x)}(\epsilon - x)d\epsilon.\end{aligned}$$

Lemma 1 gives, for our case where $\mu(x) = 0$ and $\Sigma(x) = \sigma^2(x)I_d$ for all $x \in \mathbb{R}^d$, that

$$\nabla_x \phi_{\mu(x),\Sigma(x)}(\epsilon - x) = \phi_{\mu(x),\Sigma(x)}(\epsilon - x)\left(\frac{\epsilon - x}{\sigma^2(x)} + \frac{\|\epsilon - x\|_2^2 - d\sigma^2(x)}{\sigma^3(x)}\nabla\sigma(x)\right).$$

Thus,

$$\begin{aligned}\nabla \overline{g}(x) = &\int_{\mathbb{R}^d} g(\epsilon)\frac{\epsilon - x}{\sigma^2(x)}\phi_{\mu(x),\Sigma(x)}(x - \epsilon)d\epsilon \\ &+ \int_{\mathbb{R}^d} g(\epsilon)\frac{\|\epsilon - x\|_2^2 - d\sigma^2(x)}{\sigma^3(x)}\nabla\sigma(x)\phi_{\mu(x),\Sigma(x)}(\epsilon - x)d\epsilon.\end{aligned}$$

In terms of expectations, this can be written as

$$\begin{aligned}\nabla \overline{g}(x) = &\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\sigma^2(x)I_d)}\left[\frac{\epsilon}{\sigma^2(x)}g(x+\epsilon)\right] \\ &+ \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\sigma^2(x)I_d)}\left[\frac{\nabla\sigma(x)}{\sigma^3(x)}(\|\epsilon\|_2^2 - d\sigma^2(x))g(x+\epsilon)\right].\end{aligned}$$

Since we'd like to bound the norm of this gradient, let $u \in \mathbb{R}^d$ be an arbitrary vector satisfying $\|u\|_2 = 1$. Then our goal reduces to bounding

$$\begin{aligned}u^\top \nabla \overline{g}(x) = &\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\sigma^2(x)I_d)}\left[\frac{u^\top\epsilon}{\sigma^2(x)}g(x+\epsilon)\right] \\ &+ \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\sigma^2(x)I_d)}\left[\frac{u^\top\nabla\sigma(x)}{\sigma^3(x)}(\|\epsilon\|_2^2 - d\sigma^2(x))g(x+\epsilon)\right].\end{aligned} \tag{7}$$

We now bound each of the two expectations in (7) using Lemma 2.

Applying Lemma 2 to the first expectation in (7) with $w = x$, $p = \overline{g}(x)$, $l(z) = \frac{u^\top z}{\sigma^2(x)}$, and $\nu = \mathcal{N}(\mu(x), \Sigma(x))$, we find that

$$\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\sigma^2(x)I_d)}\left[\frac{u^\top\epsilon}{\sigma^2(x)}g(x+\epsilon)\right] \le \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\sigma^2(x)I_d)}\left[\frac{u^\top\epsilon}{\sigma^2(x)}\mathbf{1}_{\{z\in\mathbb{R}^d : u^\top z/\sigma^2(x) \ge t_p\}}(\epsilon)\right],$$

where $t_p \in \mathbb{R}$ is such that $\mathbb{P}_{\epsilon\sim\mathcal{N}(\mu(x),\Sigma(x))}(u^\top\epsilon/\sigma^2(x) \ge t_p) = p$, which clearly exists since $\mathcal{N}(\mu(x), \Sigma(x))$ is a continuous distribution and hence $t \mapsto \mathbb{P}_{\epsilon\sim\mathcal{N}(\mu(x),\Sigma(x))}(u^\top\epsilon/\sigma^2(x) \ge t)$ is continuous, and has range $(0, 1)$ for $t \in \mathbb{R}$. Consider the change of variables $\epsilon' \coloneqq \epsilon/\sigma(x)$, so that $\epsilon' = \mathcal{N}(0, I_d)$, and hence $\tilde{\epsilon} \coloneqq u^\top\epsilon' \sim \mathcal{N}(0, 1)$. This change of variables shows that $p = \mathbb{P}_{\tilde{\epsilon}\sim\mathcal{N}(0,1)}(\tilde{\epsilon} \ge \sigma(x)t_p)$ and hence

$$\begin{aligned}\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\sigma^2(x)I_d)}\left[\frac{u^\top\epsilon}{\sigma^2(x)}g(x+\epsilon)\right] &\le \frac{1}{\sigma(x)}\mathbb{E}_{\tilde{\epsilon}\sim\mathcal{N}(0,1)}\left[\tilde{\epsilon}\mathbf{1}_{\{\tilde{z}\in\mathbb{R}:\tilde{z}\ge\sigma(x)t_p\}}(\epsilon)\right] \\ &= \frac{1}{\sigma(x)}\int_{\tilde{z}=\sigma(x)t_p}^{\infty}\tilde{z}\phi(\tilde{z})d\tilde{z} \\ &= \frac{1}{\sigma(x)}\phi(\sigma(x)t_p) \\ &= \frac{1}{\sigma(x)}\phi(\Phi^{-1}(p)) \\ &= \frac{1}{\sigma(x)}\phi(\Phi^{-1}(\overline{g}(x))).\end{aligned}$$

This gives our bound on the first expectation in (7).

We now bound the second expectation in (7). Applying Lemma 2 with $w = x$, $p = \overline{g}(x)$, $l(z) = \frac{u^\top \nabla\sigma(x)}{\sigma^3(x)}(\|\epsilon\|_2^2 - d\sigma^2(x))$, and $\nu = \mathcal{N}(\mu(x), \Sigma(x))$, we find that

$$\begin{aligned}
&\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\sigma^2(x)I_d)}\left[\frac{u^\top\nabla\sigma(x)}{\sigma^3(x)}(\|\epsilon\|_2^2 - d\sigma^2(x))g(x+\epsilon)\right] \\
&\quad\le \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\sigma^2(x)I_d)}\left[\frac{u^\top\nabla\sigma(x)}{\sigma^3(x)}(\|\epsilon\|_2^2 - d\sigma^2(x))\mathbf{1}_{\left\{z\in\mathbb{R}^d:\frac{u^\top\nabla\sigma(x)}{\sigma^3(x)}(\|z\|_2^2 - d\sigma^2(x))\ge t_p\right\}}(\epsilon)\right] \\
&\quad=: \alpha(x),
\end{aligned}$$

where $t_p \in \mathbb{R}$ is now defined according to the relation

$$\mathbb{P}_{\epsilon\sim\mathcal{N}(\mu(x),\Sigma(x))}\left(\frac{u^\top\nabla\sigma(x)}{\sigma^3(x)}(\|\epsilon\|_2^2 - d\sigma^2(x)) \ge t_p\right) = p,$$

which, similar to the previous expectation bound, exists due to continuity of the distribution. We now compute the value $\alpha(x)$. If $u^\top\nabla\sigma(x) = 0$, then clearly $\alpha(x) = 0$. Suppose that $u^\top\nabla\sigma(x) \neq 0$. Consider the change of variables $\epsilon' \coloneqq \|\epsilon/\sigma(x)\|_2^2$, so that $\epsilon' \sim \chi_d^2$. For all $z \in \mathbb{R}^d$, it holds that $z' \coloneqq \|z/\sigma(x)\|_2^2 \ge \epsilon_0(p)$ if and only if $l(z) = \frac{u^\top\nabla\sigma(x)}{\sigma^3(x)}(\|z\|_2^2 - d\sigma^2(x)) \ge t_p$, where

$$\epsilon_0(p) \coloneqq d + \frac{t_p\sigma(x)}{u^\top\nabla\sigma(x)}.$$

Therefore, the change of variables gives that

$$\begin{aligned}
\alpha(x) &= \mathbb{E}_{\epsilon'\sim\chi_d^2}\left[\frac{u^\top\nabla\sigma(x)}{\sigma(x)}(\epsilon' - d)\mathbf{1}_{\{z'\in\mathbb{R}:z'\ge\epsilon_0(p)\}}(\epsilon')\right] \\
&= \frac{u^\top\nabla\sigma(x)}{\sigma(x)}\int_{\epsilon'=\epsilon_0(p)}^{\infty}(\epsilon' - d)d\chi_d^2(\epsilon').
\end{aligned} \tag{8}$$

We now focus on computing this integral. First, notice that the integral of the constant term is given by

$$\int_{\epsilon'=\epsilon_0(p)}^{\infty} 1 d\chi_d^2(\epsilon') = \mathbb{P}_{\epsilon'\sim\chi_d^2}(\epsilon' \ge \epsilon_0(p)) = \mathbb{P}_{\epsilon\sim\mathcal{N}(0,\sigma^2(x)I_d)}(l(\epsilon) \ge t_p) = p. \tag{9}$$

Therefore, all that remains to compute is

$$\int_{\epsilon'=\epsilon_0(p)}^{\infty}\epsilon' d\chi_d^2(\epsilon').$$

We may assume without loss of generality that $\epsilon_0(p) \ge 0$, for otherwise it would be the case that $\mathbb{P}_{\epsilon'\sim\chi_d^2}(\epsilon' \ge \epsilon_0(p)) = 1$ and hence that $\mathbb{P}_{\epsilon\sim\mathcal{N}(0,\sigma^2 I_d)}(l(\epsilon) \ge t_p) = 1$, and therefore we could have equivalently chosen t_p such that $\epsilon_0(p) = d + \frac{t_p\sigma(x)}{u^\top\nabla\sigma(x)} = 0$ from the outset. Thus, using the χ_d^2 density function over nonnegative reals, we have that

$$\begin{aligned}
\int_{\epsilon'=\epsilon_0(p)}^{\infty}\epsilon' d\chi_d^2(\epsilon') &= \int_{\epsilon'=\epsilon_0(p)}^{\infty}\epsilon'\frac{1}{2^{d/2}\Gamma\left(\frac{d}{2}\right)}(\epsilon')^{d/2-1}e^{-\epsilon'/2}d\epsilon' \\
&= \frac{1}{2^{d/2}\Gamma\left(\frac{d}{2}\right)}\int_{\epsilon'=\epsilon_0(p)}^{\infty}(\epsilon')^{d/2}e^{-\epsilon'/2}d\epsilon'.
\end{aligned}$$

After the change of variables $\tilde{\epsilon} \coloneqq \epsilon'/2$, the last integral is given by the upper incomplete gamma function:

$$\int_{\epsilon'=\epsilon_0(p)}^{\infty}(\epsilon')^{d/2}e^{-\epsilon'/2}d\epsilon' = 2^{d/2+1}\Gamma\left(\frac{d}{2}+1, \frac{\epsilon_0(p)}{2}\right).$$

Thus, we find that

$$\int_{\epsilon'=\epsilon_0(p)}^{\infty}\epsilon' d\chi_d^2(\epsilon') = \frac{2\Gamma\left(\frac{d}{2}+1, \frac{\epsilon_0(p)}{2}\right)}{\Gamma\left(\frac{d}{2}\right)}. \tag{10}$$

Combining (8), (9), and (10) yields that

$$\alpha(x) = \frac{u^\top \nabla\sigma(x)}{\sigma(x)} \left(\frac{2\Gamma\left(\frac{d}{2}+1, \frac{\epsilon_0}{2}\right)}{\Gamma\left(\frac{d}{2}\right)} - dp \right).$$

This is simplified by noting that

$$\begin{aligned} p &= \mathbb{P}_{\epsilon' \sim \chi^2_d}(\epsilon' \geq \epsilon_0(p)) \\ &= \int_{\epsilon'=\epsilon_0(p)}^{\infty} \frac{1}{2^{d/2}\Gamma\left(\frac{d}{2}\right)} (\epsilon')^{d/2-1} e^{-\epsilon'/2} d\epsilon', \end{aligned}$$

which again can be written in terms of the upper incomplete gamma function:

$$p = \frac{\Gamma\left(\frac{d}{2}, \frac{\epsilon_0(p)}{2}\right)}{\Gamma\left(\frac{d}{2}\right)}.$$

Therefore,

$$\alpha(x) = \frac{u^\top \nabla\sigma(x)}{\sigma(x)} \frac{2\Gamma\left(\frac{d}{2}+1, \frac{\epsilon_0(p)}{2}\right) - d\Gamma\left(\frac{d}{2}, \frac{\epsilon_0(p)}{2}\right)}{\Gamma\left(\frac{d}{2}\right)}.$$

Now, the recurrence relation for the upper incomplete gamma function can be used to relate the two terms in the numerator of the second factor:

$$\begin{aligned} \alpha(x) &= \frac{u^\top \nabla\sigma(x)}{\sigma(x)} \frac{d\Gamma\left(\frac{d}{2}, \frac{\epsilon_0(p)}{2}\right) + 2\left(\frac{\epsilon_0(p)}{2}\right)^{d/2} \exp\left(-\frac{\epsilon_0(p)}{2}\right) - d\Gamma\left(\frac{d}{2}, \frac{\epsilon_0(p)}{2}\right)}{\Gamma\left(\frac{d}{2}\right)} \\ &= 2\frac{u^\top \nabla\sigma(x)}{\sigma(x)} \frac{\left(\frac{\epsilon_0(p)}{2}\right)^{d/2} \exp\left(-\frac{\epsilon_0(p)}{2}\right)}{\Gamma\left(\frac{d}{2}\right)} \\ &= 2\frac{u^\top \nabla\sigma(x)}{\sigma(x)} \frac{\epsilon_0(p)^{d/2} \exp\left(-\frac{\epsilon_0(p)}{2}\right)}{2^{d/2}\Gamma\left(\frac{d}{2}\right)} \\ &= 2\frac{u^\top \nabla\sigma(x)}{\sigma(x)} \epsilon_0(p) f_{\chi^2_d}(\epsilon_0(p)). \end{aligned}$$

This gives our bound on the second expectation in (7).

Applying our established bounds on the expectations in (7), we find that

$$u^\top \nabla\overline{g}(x) \leq \frac{1}{\sigma(x)} \phi(\Phi^{-1}(\overline{g}(x))) + 2\frac{u^\top \nabla\sigma(x)}{\sigma(x)} \epsilon_0(p) f_{\chi^2_d}(\epsilon_0(p)).$$

Now, we see that

$$p = \mathbb{P}_{\epsilon' \sim \chi^2_d}(\epsilon' \geq \epsilon_0(p)) = 1 - \mathbb{P}_{\epsilon' \sim \chi^2_d}(\epsilon' < \epsilon_0(p)) = 1 - F_{\chi^2_d}(\epsilon_0(p)),$$

where $F_{\chi^2_d}$ is the cumulative distribution function of the χ^2_d distribution, and hence

$$\epsilon_0(p) = F^{-1}_{\chi^2_d}(1-p).$$

This shows, importantly, that $\epsilon_0(p)$ is actually independent of u. Based on this key observation, we conclude that

$$\begin{aligned} \|\nabla\overline{g}(x)\|_2 &= \sup_{u \in \mathbb{R}^d : \|u\|_2 = 1} u^\top \nabla\overline{g}(x) \\ &\leq \frac{1}{\sigma(x)} \phi(\Phi^{-1}(\overline{g}(x))) + \sup_{u \in \mathbb{R}^d : \|u\|_2 = 1} 2\frac{u^\top \nabla\sigma(x)}{\sigma(x)} \epsilon_0(p) f_{\chi^2_d}(\epsilon_0(p)) \\ &= \frac{1}{\sigma(x)} \phi(\Phi^{-1}(\overline{g}(x))) + 2\frac{\|\nabla\sigma(x)\|_2}{\sigma(x)} \epsilon_0(p) f_{\chi^2_d}(\epsilon_0(p)). \end{aligned}$$

Therefore, (6) gives that

$$\begin{aligned}\|\nabla_x \Phi^{-1}(\overline{g}(x))\|_2 &= \frac{\|\nabla \overline{g}(x)\|_2}{\phi(\Phi^{-1}(\overline{g}(x)))} \\ &\leq \frac{1}{\sigma(x)} + 2\frac{\|\nabla \sigma(x)\|_2}{\sigma(x)} \frac{\epsilon_0(p) f_{\chi^2_d}(\epsilon_0(p))}{\phi(\Phi^{-1}(p))} \\ &\leq \frac{1}{\sigma_{\min}} \left(1 + 2\,\mathrm{Lip}(\sigma) \frac{\epsilon_0(p) f_{\chi^2_d}(\epsilon_0(p))}{\phi(\Phi^{-1}(p))}\right).\end{aligned}$$

Since $p' \mapsto \frac{\epsilon_0(p') f_{\chi^2_d}(\epsilon_0(p'))}{\phi(\Phi^{-1}(p'))}$ is monotone decreasing on $[p_{\min}, 1 - p_{\min}]$ by Lemma 3, and since $p = \overline{g}(x) \in [p_{\min}, 1 - p_{\min}]$, we conclude that

$$\|\nabla_x \Phi^{-1}(\overline{g}(x))\|_2 \leq \frac{1}{\sigma_{\min}} \left(1 + 2\,\mathrm{Lip}(\sigma) \frac{\epsilon_0(p_{\min}) f_{\chi^2_d}(\epsilon_0(p_{\min}))}{\phi(\Phi^{-1}(p_{\min}))}\right) = L,$$

and therefore indeed $x \mapsto \Phi^{-1}(\overline{g}(x))$ is L-Lipschitz continuous, as x is arbitrary. □

Theorem 3. *Consider input-dependent smoothing, with all hypotheses of Theorem 2 satisfied; $\mu(x) = 0$, $\Sigma(x) = \sigma^2(x) I_d$ with σ Lipschitz continuous and bounded below by $\sigma_{\min} > 0$, and $\overline{g}(x) \in [p_{\min}, 1 - p_{\min}]^n$ for $p_{\min} \in (0, 1/2)$ as in Lemma 3. Consider a point $x \in \mathbb{R}^d$, let $y = \overline{f}(x)$ be the classification of x under the smoothed classifier, and let $y' \in \arg\max_{i \in \{1,\ldots,n\} \setminus \{y\}} \overline{g}_i(x)$ be the runner-up class. Then, it holds that*

$$\overline{f}(x + \delta) = y$$

for all $\delta \in \mathbb{R}^d$ such that

$$\|\delta\|_2 \leq \overline{r}(x, y) \coloneqq \frac{1}{2L} \left(\Phi^{-1}(\overline{g}_y(x)) - \Phi^{-1}(\overline{g}_{y'}(x))\right),$$

with L being the Lipschitz constant defined in (5)*.*

Proof of Theorem 3. The proof uses the same argument as in Zhai et al. [17, Theorem 2], with the Lipschitz constant L from Theorem 2. □