Double Sampling Randomized Smoothing

Linyi Li [1] **Jiawei Zhang** [1] **Tao Xie** [2] **Bo Li** [1]

Abstract

Neural networks (NNs) are known to be vulnerable against adversarial perturbations, and thus there is a line of work aiming to provide robustness certification for NNs, such as randomized smoothing, which samples smoothing noises from a certain distribution to certify the robustness for a smoothed classifier. However, as shown by previous work, the certified robust radius in randomized smoothing suffers from scaling to large datasets ("curse of dimensionality"). To overcome this hurdle, we propose a **D**ouble **S**ampling **R**andomized **S**moothing (**DSRS**) framework, which exploits the sampled probability from an *additional smoothing distribution* to tighten the robustness certification of the previous smoothed classifier. Theoretically, under mild assumptions, we prove that DSRS can certify $\Theta(\sqrt{d})$ robust radius under ℓ_2 norm where d is the input dimension, implying that *DSRS may be able to break the curse of dimensionality of randomized smoothing*. We instantiate DSRS for a generalized family of Gaussian smoothing and propose an efficient and sound computing method based on customized dual optimization considering sampling error. Extensive experiments on MNIST, CIFAR-10, and ImageNet verify our theory and show that DSRS certifies larger robust radii than existing baselines consistently under different settings. Code is available at `https://github.com/llylly/DSRS`.

1. Introduction

Neural networks (NNs) have achieved great advances on a wide range of tasks, but have been shown vulnerable against adversarial examples (Szegedy et al., 2014; Goodfellow et al., 2015; Eykholt et al., 2018; Wang et al., 2021; Qiu et al., 2020; Li et al., 2020a; Zhang et al., 2021). A plethora of empirical defenses are proposed to improve the robustness; however, most of these are broken by strong adversaries again (Carlini & Wagner, 2017; Athalye et al., 2018; Tramèr et al., 2020). Recently, there are great efforts in developing certified defenses for NNs under certain adversarial constraints (Wong & Kolter, 2018; Raghunathan et al., 2018; Li et al., 2020b; Xu et al., 2020; Li et al., 2019b).

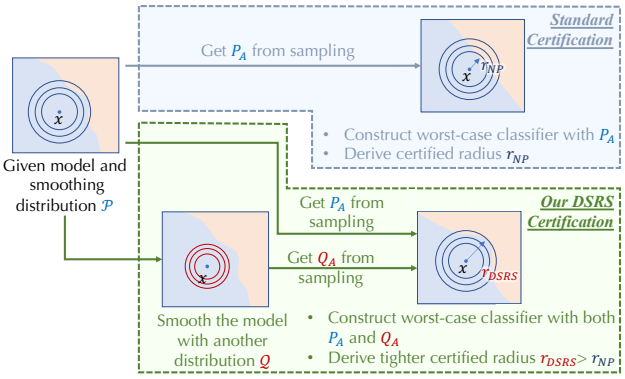


Figure 1. **Upper**: Standard certification for randomized smoothing leverages information from only one distribution (P_A) to compute robustness certification. **Lower**: DSRS leverages information from two distributions (P_A and Q_A) to compute certification for the smoothed classifier, yielding significantly larger certified radius.

Randomized smoothing (Cohen et al., 2019; Li et al., 2019a) has emerged as a popular technique to provide certified robustness for large-scale datasets. Concretely, it samples noise from a certain smoothing distribution to construct a smoothed classifier, and thus certifies the robust radius for the smoothed classifier. Compared to other techniques (Wong & Kolter, 2018; Mirman et al., 2018; Gowal et al., 2019; Zhang et al., 2020b), randomized smoothing is efficient and agnostic to the model, and is applicable to a wide range of ML models, including large ResNet (He et al., 2016) on the ImageNet dataset.

To improve the certified robust radius, existing studies (Cohen et al., 2019; Lee et al., 2019; Li et al., 2019a; Yang et al., 2020) have explored different smoothing distributions. However, the improvement is limited. For example, ℓ_2 certified robust radius does not increase on large datasets despite that the input dimension d increases (Cohen et al., 2019), resulting in a low ℓ_∞ certified radius on large datasets, theo-

[1]University of Illinois Urbana-Champaign, Illinois, USA [2]Peking University, Beijing, China. Correspondence to: Linyi Li <linyi2@illinois.edu>, Tao Xie <taoxie@pku.edu.cn>, Bo Li <lbo@illinois.edu>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

retically shown as an intrinsic barrier of randomized smoothing ("curse of dimensionality" or "ℓ_∞ barrier") (Yang et al., 2020; Blum et al., 2020; Kumar et al., 2020b; Hayes, 2020; Wu et al., 2021).

Given these challenges toward tight robustness certification, a natural question arises: ***Q1**: Is it possible to circumvent the barrier of randomized smoothing by certifying with additional "information"? **Q2**: What type of information is needed to provide tight robustness certification?* To answer these questions, we propose a **D**ouble **S**ampling **R**andomized **S**moothing (DSRS) framework to leverage the sampled noises from an *additional smoothing distribution* as additional information to tighten the robust certification. In theory, we show that (1) ideally, if the decision region of the base classifier is known, DSRS can provide tight robustness certification; (2) more practically, if the inputs, which can be correctly classified by the base classifier, satisfy the concentration property within an input-centered ball with constant mass under standard Gaussian measure, the standard Neyman-Pearson-based certification (Li et al., 2019a; Cohen et al., 2019; Salman et al., 2019; Yang et al., 2020) can certify only a dimension-independent ℓ_2 radius, whereas DSRS with generalized Gaussian smoothing can certify radius $\Omega(\sqrt{d})$ (under ℓ_2 norm), which would increase with the dimension d, leading to tighter certification. Under more general conditions, we provide numerical simulations to verify our theory. Our results provide a positive answer to Q1 and sufficient conditions for Q2, i.e., DSRS may be able to circumvent the barrier of randomized smoothing.

Motivated by the theory, we leverage a type of generalized Gaussian (Zhang et al., 2020a) as the smoothing distribution and truncated generalized Gaussian as an additional distribution. For this type of concretization, we propose an efficient and sound computation method to compute the certifiably robust radius for practical classifiers considering sampling error. Our method formulates the certification problem given additional information as a constrained optimization problem and leverages specific properties of the dual problem to decompose the effects of different dual variables to solve it. DSRS is fully scalable since the computational time is nearly independent of the size of the dataset, model, or sampling. Our extensive experimental evaluation on MNIST, CIFAR-10, and ImageNet shows that (1) under large sampling size ($2 \times 10^5 - 8 \times 10^5$), the certified radius of DSRS consistently increases as suggested by our theory; (2) under practical sampling size (10^5), DSRS can certify consistently higher robust radii than existing baselines, including standard Neyman-Pearson-based certification.

As further discussed in Appendix L, we believe that DSRS as a framework opens a wide range of future directions for selecting or optimizing different forms of *additional information* to tighten the certification of randomized smoothing.

We summarize the main technical *contributions* as follows:

- We propose a general robustness certification framework DSRS, which leverages additional information by sampling from another smoothing distribution.
- We prove that under practical concentration assumptions, DSRS certifies $\Omega(\sqrt{d})$ radius under ℓ_2 norm with d the input dimension, suggesting a possible way to circumvent the intrinsic barrier of randomized smoothing.
- We concretize DSRS by generalized Gaussian smoothing mechanisms and propose a method to efficiently compute the certified radius for given classifiers.
- We conduct extensive experiments, showing that DSRS provides consistently tighter robustness certification than existing baselines, including standard Neyman-Pearson-based certification across different models on MNIST, CIFAR-10, and ImageNet.

Related Work. For the certification method of randomized smoothing, most existing methods leverage only the true-class prediction probability to certify. In this case, the tightest possible robustness certification is based on the Neyman-Pearson lemma (Neyman & Pearson, 1933) as first proposed by Cohen et al. (2019) for certifying ℓ_2 radius under Gaussian smoothing. Several methods extend this certification to accommodate different smoothing distributions and different ℓ_p norms (Dvijotham et al., 2020; Yang et al., 2020; Zhang et al., 2020a; Levine & Feizi, 2021). In randomized smoothing, the ℓ_2 certified robust radius r is similar across datasets of different scales, resulting in the vanishing ℓ_∞ certified radius $r/\sqrt{d}$ when input dimension increases. This limitation of existing certification methods of randomized smoothing is formally proved (Yang et al., 2020; Blum et al., 2020; Kumar et al., 2020b; Hayes, 2020; Wu et al., 2021) and named "ℓ_∞ barrier" or "curse of dimensionality".

Recent work tries to incorporate additional information besides true-class prediction probability to tighten the certification and bypass the barrier. For ℓ_2 and ℓ_∞ certification, to the best of our knowledge, gradient magnitude is the only exploited additional information (Mohapatra et al., 2020; Levine et al., 2020). However, in practice, the improvement is relatively marginal and requires a large number of samples (see Appendix J.5). Some other methods provide tighter certification given specific model structures (Kumar et al., 2020a; Chiang et al., 2020; Lee et al., 2019; Awasthi et al., 2020). DSRS instead focuses on leveraging model-structure-agnostic additional information. More discussion on related work is in Appendix K.

2. Preliminaries and Background

Define $[C] := \{1, \dots C\}$. Let $\mathbf{\Delta}^C$ be the C-dimensional probability simplex. We consider a multiclass classification

model $F : \mathbb{R}^d \to [C]$ as the *base classifier*, where d is the input dimension, and the model outputs hard-label class prediction within $[C]$. The *original smoothing distribution* $\mathcal{P}$ and *additional smoothing distribution* $\mathcal{Q}$ are both supported on $\mathbb{R}^d$. We let $p(\cdot)$ and $q(\cdot)$ be their density functions respectively. We assume that both p and q are positive and differentiable almost everywhere, i.e., the set of singular points has zero measure under either $\mathcal{P}$ or $\mathcal{Q}$. These assumptions hold for common smoothing distributions used in the literature such as Gaussian distribution (Lécuyer et al., 2019; Li et al., 2019a; Cohen et al., 2019; Yang et al., 2020).

Randomized smoothing constructs a smoothed classifier from a given base classifier by adding input noise following *original* smoothing distribution $\mathcal{P}$. For input $\boldsymbol{x} \in \mathbb{R}^d$, we define *prediction probability under* $\mathcal{P}$ by function $f^{\mathcal{P}} : \mathbb{R}^d \to \boldsymbol{\Delta}^C$:

$$f^{\mathcal{P}}(\boldsymbol{x})_c := \Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[F(\boldsymbol{x}+\boldsymbol{\epsilon}) = c] \quad \text{where} \quad c \in [C]. \tag{1}$$

The *smoothed classifier* $\widetilde{F}^{\mathcal{P}} : \mathbb{R} \to [C]$ (or $\widetilde{F}$ when $\mathcal{P}$ is clear from the context) predicts the class with the highest confidence after smoothing with $\mathcal{P}$:

$$\widetilde{F}^{\mathcal{P}}(\boldsymbol{x}) := \arg\max_{c\in[C]} f^{\mathcal{P}}(\boldsymbol{x})_c. \tag{2}$$

We focus on robustness certification against ℓ_p-bounded perturbations for smoothed classifier $\widetilde{F}$, where the standard certification method is called Neyman-Pearson-based certification (Cohen et al., 2019) (details in Appendix A, certified radius from it denoted by $r_{\mathsf{N-P}}$). Concretely, certification methods compute robust radius r defined as below.

Definition 1 (Certified Robust Radius)**.** Under ℓ_p norm ($p \in \mathbb{R}_+ \cup \{+\infty\}$), for given smoothed classifier $\widetilde{F}^{\mathcal{P}}$ and input $\boldsymbol{x}_0 \in \mathbb{R}^d$ with true label $y_0 \in [C]$, a radius $r \geq 0$ is called *certified (robust) radius* for $\widetilde{F}^{\mathcal{P}}$ if $\widetilde{F}^{\mathcal{P}}$ always predicts y_0 for any input within the r-radius ball centered at $\boldsymbol{x}_0$:

$$\forall \boldsymbol{\delta} \in \mathbb{R}^d, \|\boldsymbol{\delta}\|_p < r, \widetilde{F}^{\mathcal{P}}(\boldsymbol{x}_0+\boldsymbol{\delta}) = y_0. \tag{3}$$

3. DSRS Overview

We propose **D**ouble **S**ampling **R**andomized **S**moothing (DSRS), which leverages the prediction probability from an *additional* smoothing distribution $\mathcal{Q}$ (formally $Q_A := f^{\mathcal{Q}}(\boldsymbol{x}_0)_{y_0} = \Pr_{\boldsymbol{\epsilon}\sim\mathcal{Q}}[F(\boldsymbol{x}+\boldsymbol{\epsilon}) = y_0]$), along with the prediction probability from the original smoothing distribution $\mathcal{P}$ (formally $P_A := f^{\mathcal{P}}(\boldsymbol{x}_0)_{y_0}$ as in Eqn. (1), also used in Neyman-Pearson-based certification), to provide robustness certification for $\mathcal{P}$-smoothed classifier $\widetilde{F}^{\mathcal{P}}$. Note that both P_A and Q_A can be obtained from Monte-Carlo sampling (see Sections 5.1 and 5.2). Formally, we let r_{DSRS} denote the tightest possible certified radius with prediction probability from $\mathcal{Q}$, then r_{DSRS} can be defined as below.

Definition 2 (r_{DSRS})**.** Given P_A and Q_A,

$$\begin{aligned} &r_{\mathsf{DSRS}} := \max r \quad \text{s.t.} \\ &\forall F : \mathbb{R} \to [C], f^{\mathcal{P}}(\boldsymbol{x}_0)_{y_0} = P_A, f^{\mathcal{Q}}(\boldsymbol{x}_0)_{y_0} = Q_A \\ \Rightarrow &\forall \boldsymbol{x}, \|\boldsymbol{x}-\boldsymbol{x}_0\|_p < r, \widetilde{F}^{\mathcal{P}}(\boldsymbol{x}) = y_0. \end{aligned} \tag{4}$$

Table 1. Definitions of smoothing distributions in this paper. In the table, $k \in \mathbb{N}$, $\sigma' = \sqrt{d/(d-2k)}\sigma$.

Name	Notation	Density Function
Standard Gaussian	$\mathcal{N}(\sigma)$	$\propto \exp\left(-\frac{\|\boldsymbol{\epsilon}\|_2^2}{2\sigma^2}\right)$
Generalized Gaussian	$\mathcal{N}^{\mathrm{g}}(k,\sigma)$	$\propto \|\boldsymbol{\epsilon}\|_2^{-2k} \exp\left(-\frac{\|\boldsymbol{\epsilon}\|_2^2}{2\sigma'^2}\right)$
Truncated Standard Gaussian	$\mathcal{N}_{\mathsf{trunc}}(T,\sigma)$	$\propto \exp\left(-\frac{\|\boldsymbol{\epsilon}\|_2^2}{2\sigma^2}\right) \cdot \mathbb{I}[\|\boldsymbol{\epsilon}\|_2 \leq T]$
Truncated Generalized Gaussian	$\mathcal{N}^{\mathrm{g}}_{\mathsf{trunc}}(k,T,\sigma)$	$\propto \|\boldsymbol{\epsilon}\|_2^{-2k} \exp\left(-\frac{\|\boldsymbol{\epsilon}\|_2^2}{2\sigma'^2}\right) \cdot \mathbb{I}[\|\boldsymbol{\epsilon}\|_2 \leq T]$

Intuitively, r_{DSRS} is the maximum possible radius, such that any smoothed classifier constructed from base classifier satisfying P_A and Q_A constraints cannot predict other labels when the perturbation magnitude is within the radius.

In Section 4, we will analyze the theoretical properties of DSRS, including comparing r_{DSRS} and $r_{\mathsf{N-P}}$ under the concentration assumption. Computing r_{DSRS} is nontrivial, so in Section 5, we will introduce a practical computational method that exactly solves r_{DSRS} when $\mathcal{P}$ and $\mathcal{Q}$ are standard and generalized (truncated) Gaussian. In Appendix G, we will show method variants to deal with other forms of $\mathcal{P}$ and $\mathcal{Q}$ distributions. In Appendix L, we will further generalize the DSRS framework.

Smoothing Distributions. Now we formally define the smoothing distributions used in DSRS. We mainly consider standard Gaussian $\mathcal{N}$ (Cohen et al., 2019; Yang et al., 2020) and generalized Gaussian $\mathcal{N}^{\mathrm{g}}$ (Zhang et al., 2020a). Let $\mathcal{N}(\sigma)$ to represent standard Gaussian distribution with covariance matrix $\sigma^2\boldsymbol{I}_d$ that has density function $\propto \exp(-\|\boldsymbol{\epsilon}\|_2^2/(2\sigma^2))$.[1] For $k \in \mathbb{N}$, we let $\mathcal{N}^{\mathrm{g}}(k,\sigma)$ to represent generalized Gaussian whose density function $\propto \|\boldsymbol{\epsilon}\|_2^{-2k} \exp(-\|\boldsymbol{\epsilon}\|_2^2/(2\sigma'^2))$ where $\sigma' = \sqrt{d/(d-2k)}\sigma$. Here we use σ' instead of σ to ensure that the expected noise $\sqrt{\mathbb{E}\|\boldsymbol{\epsilon}\|_2^2}$ of $\mathcal{N}^{\mathrm{g}}(k,\sigma)$ is the same as $\mathcal{N}(\sigma)$. The generalized Gaussian as the smoothing distribution overcomes the "thin shell" problem of standard Gaussian and improves certified robustness (Zhang et al., 2020a); and we will reveal more of its theoretical advantages in Section 4.

As the additional smoothing distribution $\mathcal{Q}$, we will mainly consider truncated distributions within a small ℓ_2 radius ball. Specially, truncated standard Gaussian is denoted by $\mathcal{N}_{\mathsf{trunc}}(T,\sigma)$ with density function $\propto \exp(-\|\boldsymbol{\epsilon}\|_2^2/(2\sigma^2)) \cdot \mathbb{I}[\|\boldsymbol{\epsilon}\|_2 \leq T]$; and truncated generalized Gaussian is denoted by $\mathcal{N}^{\mathrm{g}}_{\mathsf{trunc}}(k,T,\sigma)$ with density function $\propto \|\boldsymbol{\epsilon}\|_2^{-2k} \exp(-\|\boldsymbol{\epsilon}\|_2^2/(2\sigma'^2)) \cdot \mathbb{I}[\|\boldsymbol{\epsilon}\|_2 \leq T]$.

In Table 1, we summarize these distribution definitions.

[1] In this paper, $\mathcal{N}(\sigma)$ is a shorthand of $\mathcal{N}(0, \sigma^2\boldsymbol{I}_d)$.

4. Theoretical Analysis of DSRS

In this section, we theoretically analyze DSRS to answer the following core question: *Does Q_A, the prediction probability under additional smoothing distribution, provide sufficient information for tightening the robustness certification?* We first show that if the support of $\mathcal{Q}$ is the decision region of true class, DSRS can certify the smoothed classifier's maximum possible robust radius. Then, under concentration assumption, we show the ℓ_2 certified radius of DSRS can be $\Omega(\sqrt{d})$ that is asymptotically optimal for bounded inputs. Finally, under more general conditions, we conduct both numerical simulations and real-data experiments to verify that the certified radius of DSRS increases with data dimension d. These analyses provide a positive answer to the above core question.

DSRS can certify the tightest possible robust radius.
Given an original smoothing distribution $\mathcal{P}$ and a base classifier F_0. At input point $\boldsymbol{x}_0 \in \mathbb{R}^d$ with true label y_0, we define the tightest possible certified robust radius r_{tight} to be the largest ℓ_p ball that contains no adversarial example for *smoothed* classifier $\widetilde{F}_0^{\mathcal{P}}$:

$$r_{\mathsf{tight}} := \max r \,\text{s.t.}\, \forall \boldsymbol{\delta} \in \mathbb{R}^d, \|\boldsymbol{\delta}\|_p < r, \widetilde{F}_0^{\mathcal{P}}(\boldsymbol{x}_0 + \boldsymbol{\delta}) = y_0.$$

Then, for binary classification, if we choose an additional smoothing distribution $\mathcal{Q}$ whose support is the decision region or its complement, then DSRS can certify robust radius r_{tight}.

Theorem 1. *Suppose the original smoothing distribution $\mathcal{P}$ has non-zero density everywhere, i.e., $p(\cdot) > 0$. For binary classification with base classifier F_0, at point $\boldsymbol{x}_0 \in \mathbb{R}^d$, let $\mathcal{Q}$ be an additional distribution that satisfies: (1) its support is the decision region of an arbitrary class $c \in [C]$ shifted by $\boldsymbol{x}_0$: $\mathrm{supp}(\mathcal{Q}) = \{\boldsymbol{x} - \boldsymbol{x}_0 : F_0(\boldsymbol{x}) = c\}$; (2) for any $\boldsymbol{x} \in \mathrm{supp}(\mathcal{Q})$, $0 < q(\boldsymbol{x})/p(\boldsymbol{x}) < +\infty$. Then, plugging $P_A = f_0^{\mathcal{P}}(\boldsymbol{x}_0)_c$ and $Q_A = f_0^{\mathcal{Q}}(\boldsymbol{x}_0)_c$ (see Eqn. (1)) into Definition 2, we have $r_{\mathsf{DSRS}} = r_{\mathsf{tight}}$ under any ℓ_p ($p \geq 1$).*

Proof sketch. We defer the proof to Appendix F.1. At a high level, with this type of $\mathcal{Q}$, we have $Q_A = 1$ or $Q_A = 0$. Then, from the mass of the $\mathcal{Q}$'s support on $\mathcal{P}$ and P_A, we can conclude that the $\mathcal{Q}$'s support is exactly the decision region of label c or its complement. Thus, the DSRS constraints (in Eqn. (4)) are satisfied iff F differs from F_0 in a zero-measure set, and thus we exactly compute the smoothed classifier $\widetilde{F}_0^{\mathcal{P}}$'s maximum certified robust radius in DSRS. An extension to multiclass setting is in Appendix F.2. □

Remark. For any base classifier F_0, $\mathcal{Q}$ that satisfies conditions in Theorem 1 exists, implying that with DSRS, certifying a strictly tight robust radius is possible. In contrast, Neyman-Pearson-based is proved to certify tight robust radius for linear base classifiers (Cohen et al., 2019, Section 3.1), but for arbitrary base classifiers, its tightness is not guaranteed. This result suggests that, to certify a tight radius, just one additional smoothing distribution $\mathcal{Q}$ is sufficient rather than multiple ones.
On the other hand, it is challenging to find $\mathcal{Q}$ whose support (or its complement) exactly matches the decision region of an NN classifier. In the following, we analyze the tightness of DSRS under weaker assumptions.

DSRS can certify $\Omega(\sqrt{d})$ ℓ_2 radius under concentration assumption.
We begin by defining the concentration property.

Definition 3 ((σ, P_{con})-Concentration)**.** Given a base classifier F_0, at input $\boldsymbol{x}_0 \in \mathbb{R}$ with true label y_0, we call F_0 satisfies $(\sigma, P_{\mathsf{con}})$-concentration property, if for within P_{con}-percentile of small ℓ_2 magnitude Gaussian $\mathcal{N}(\sigma)$ noise, the adversarial example occupies zero measure. Formally, $(\sigma, P_{\mathsf{con}})$-concentration means

$$\Pr_{\boldsymbol{\epsilon}\sim\mathcal{N}(\sigma)}[F_0(\boldsymbol{x}_0 + \boldsymbol{\epsilon}) = y_0 \,|\, \|\boldsymbol{\epsilon}_0\|_2 \leq T] = 1 \quad (5a)$$

$$\text{where} \quad T \,\text{satisfies} \Pr_{\boldsymbol{\epsilon}\sim\mathcal{N}(\sigma)}[\|\boldsymbol{\epsilon}\|_2 \leq T] = P_{\mathsf{con}}. \quad (5b)$$

Intuitively, $(\sigma, P_{\mathsf{con}})$-concentration implies that the base classifier has few adversarial examples for small magnitude noises during standard Gaussian smoothing. In Figure 4 (in Appendix B), we empirically verified that a well-trained base classifier on ImageNet may satisfy this property for a significant portion of inputs. Furthermore, Salman et al. (2019) show that promoting this concentration property by adversarially training the smoothed classifier improves the certified robustness. With this concentration property, DSRS certifies the radius $\Omega(\sqrt{d})$ under ℓ_2 norm, as the following theorem shows.

Theorem 2. *Let d be the input dimension and F_0 be the base classifier. For an input point $\boldsymbol{x}_0 \in \mathbb{R}^d$ with true class y_0, suppose F_0 satisfies $(\sigma, P_{\mathsf{con}})$-Concentration property. Then, for any sufficiently large d, for the classifier $\widetilde{F}_0^{\mathcal{P}'}$ smoothed by generalized Gaussian $\mathcal{P}' = \mathcal{N}^{\mathsf{g}}(k, \sigma)$ with $d/2 - 15 \leq k < d/2$, DSRS with additional smoothing distribution $\mathcal{Q} = \mathcal{N}^{\mathsf{g}}_{\mathsf{trunc}}(k, T, \sigma)$ can certified ℓ_2 radius*

$$r_{\mathsf{DSRS}} \geq 0.02\sigma\sqrt{d} \quad (6)$$

where $T = \sigma\sqrt{2\Gamma\mathrm{CDF}^{-1}_{d/2}(P_{\mathsf{con}})}$ and $\Gamma\mathrm{CDF}_{d/2}$ is the CDF of gamma distribution $\Gamma(d/2, 1)$.

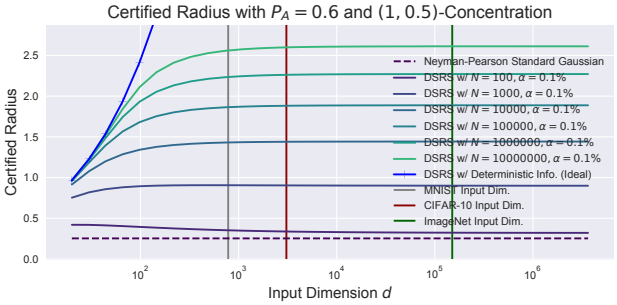
Proof sketch. We defer the proof to Appendix F.3. At high level, based on the standard Gaussian distribution's property (Proposition F.1), we find $Q_A = 1$ under concentration property (Lemma F.2). With $Q_A = 1$, we derive a lower bound of r_{DSRS} in Lemma F.3. We then use: (1) the concentration of beta distribution $\mathrm{Beta}(\frac{d-1}{2}, \frac{d-1}{2})$ (see Lemma F.4) for large d; (2) the relative concentration of gamma $\Gamma(d/2, 1)$ distribution around mean for large d (see Proposition F.5 and resulting Fact F.7); and (3) the misalignment of gamma distribution $\Gamma(d/2 - k, 1)$'s mean and median for small $(d/2 - k)$ (see Proposition F.6) to lower

bound the quantity in Lemma F.3 and show it is large or equal to 0.5. Then, using the conclusion in Section 5 we conclude that $r_{\mathsf{DSRS}} \geq 0.02\sigma\sqrt{d}$. □

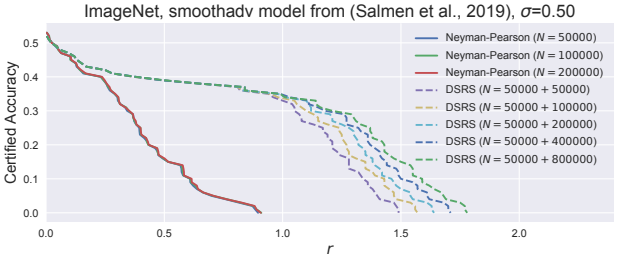
Remark. (1) For standard Neyman-Pearson based certificaton, $r_{\mathsf{N-P}} = \sigma\Phi^{-1}(f^{\mathcal{P}}(\boldsymbol{x}_0)_{y_0})$. Along with the increase of input dimension d, to achieve growing ℓ_2 certified radius, one needs the prediction probability of true class under $\mathcal{P}$, namely $f^{\mathcal{P}}(\boldsymbol{x}_0)_{y_0}$, to grow simultaneously, which is challenging. Indeed, across different datasets, $f^{\mathcal{P}}(\boldsymbol{x}_0)_{y_0}$ is almost a constant, which leads to a constant ℓ_2 certified radius and shrinking ℓ_∞ radius for large d. We further empirically illustrate this property in Appendix C. (2) In contrast, as long as the model satisfies concentration property, which may be almost true on large datasets as reflected by Figure 4, with our specific choices of $\mathcal{P}$ and $\mathcal{Q}$, DSRS can achieve $\Omega(\sigma\sqrt{d})$ ℓ_2 radius on large datasets. This rate translates to a constant $\Omega(\sigma)$ ℓ_∞ radius on large datasets and thus breaks the curse of dimensionality of randomized smoothing. We remark that this $\sqrt{d}$ rate is optimal when dataset input is bounded such as images (otherwise, the $\omega(1)$ ℓ_∞ radius leads the radius to exceed the constant ℓ_∞ diameter for large d). Therefore, *under the assumption of concentration property, DSRS provides asymptotically optimal certification for randomized smoothing*. (3) Smoothing with generalized Gaussian distribution and choosing a parameter k that is close to $d/2$ play an essential role in proving the $\Omega(\sigma\sqrt{d})$ certified radius. Otherwise, in Appendix F.4 we have Theorem 6 that shows any certification methods cannot certify an ℓ_2 radius $c\sqrt{d}$ for any $c > 0$. This adds another theoretical evidence for the superiority of generalized Gaussian that is cross-validated by Zhang et al. (2020a).

DSRS certifies tighter radius under general scenarios. When the concentration property does not absolutely hold, a rigorous theoretical analysis becomes challenging, since the impact of a noninfinite dual variable needs to be taken into account. This dual variable is inside a Lambert W function where typical approximation bounds are too loose to provide non-trivial convergence rates. Thus, we leverage the numerical computational method introduced in Section 5 to provide numerical simulations and real-data experiments. We generalize the concentration assumption by changing the holding probability in Eqn. (5a) from 1 to $\alpha^{1/N}$, which corresponds to $(1-\alpha)$-confident lower bound of Q_A given N times of Monte-Carlo sampling, where we set $\alpha = 0.1\%$ following the convention (Cohen et al., 2019). In this scenario, we compare DSRS certification with Neyman-Pearson certification numerically in Figure 2 (numerical simulations in Figure 2(a) and ImageNet experiments in Figure 2(b)).

In Figure 2(a), we assume $(\sigma, P_{\mathsf{con}})$-concentration with $\sigma = 1$, $P_{\mathsf{con}} = 0.5$ and different sampling number Ns. We further assume $P_A = f^{\mathcal{P}}(\boldsymbol{x}_0)_{y_0} = 0.6$ as the true-class prediction probability under $\mathcal{P}$. In Figure 2(b), we take the model weights trained by Salman et al. (2019) on ImageNet and apply generalized Gaussian smoothing with $d/2-k = 4$ and $\sigma = 0.50$. We uniformly pick 100 samples from the test set and compute $(1-\alpha)$-confident certified radius for each sample. We report certified accuracy (under different ℓ_2 radius r) that is the fraction of certifiably correctly classified samples by the smoothed classifier.



(a) When holding probability in Eqn. (5a) is obtained from sampling N times and confidence level $1-\alpha = 99.9\%$, relation between certified radius (y-axis) and input data dimension d (x-axis). Different curves correspond to different Ns.



(b) Relation between certified radius (x-axis) and certified accuracy (y-axis) on ImageNet models. Different curves correspond to Neyman-Pearson and DSRS certification with different Ns. Sampling error considered, confidence level $= 99.9\%$.

Figure 2. Tendency of DSRS certified robust radius considering sampling error. In both **(a)** and **(b)**, DSRS certified radius grows along with the increase of sampling number N but Neyman-Pearson radius is almost fixed.

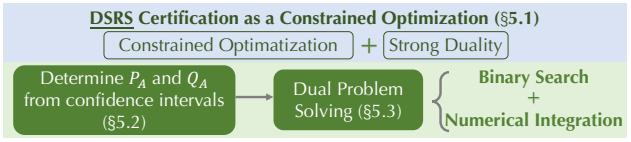


Figure 3. Overview of DSRS computational method.

Remark. When the sampling error and confidence interval come into play, they quickly suppress the $\Omega(\sqrt{d})$ growth rate of DSRS certified radius (blue curve) as shown in Figure 2(a). Nonetheless, DSRS still certifies a larger radius than the standard Neyman-Pearson method and increasing the sampling number further enlarges the gap.

We consider another relaxed version of concentration property in Appendix D, where DSRS still provides significantly tighter robustness certification than Neyman-Pearson.

5. DSRS Computational Method

The theoretical analysis in Section 4 implies that *additional smoothing distribution* $\mathcal{Q}$ helps to tighten the robustness certification over standard Neyman-Pearson-based certification significantly. In this section, we propose an efficient compu-

tational method to compute this tight certified robust radius r_{DSRS} (see Definition 2) when $\mathcal{P}$ is generalized Gaussian and $\mathcal{Q}$ is truncated $\mathcal{P}$ as suggested by Theorems 2 and 6.

Compared with the classical certification for randomized smoothing or its variants (cf. (Kumar et al., 2020a)), incorporating additional information raises a big challenge: the Neyman-Pearson lemma (1933) can no longer be served as the foundation of the certification algorithm due to its incapability to handle the additional information.

Thus, we propose a novel DSRS computational method by formalizing robustness certification as a constrained optimization problem and proving its strong duality (§5.1). Then, we propose an efficient algorithm to solve this specific dual optimization problem considering sampling error. The detailed algorithm can be found in Alg. 2 in Appendix E.1: 1) we first perform a binary search on the certified radius r to determine the maximum radius that we can certify; 2) for current r, we determine the smoothed prediction confidence P_A and Q_A from the confidence intervals of predicting the true class (§5.2); 3) then, for current r we solve the dual problem by quick binary search for dual variables λ_1 and λ_2 (see Eqn. (10)) along with numerical integration (§5.3). To guarantee the soundness of numerical-integration-based certification, we take the maximum possible error into account during the binary search. We will discuss further extensions in §5.4.

5.1. DSRS as Constrained Optimization

We first formulate the robustness certification as a constrained optimization problem and then show several foundational properties of the problem.

Following the notation of Definition 2, from the given base classifier F_0, we can use Monte-Carlo sampling to obtain

$$P_A = f_0^{\mathcal{P}}(\boldsymbol{x}_0)_{y_0}, \quad Q_A = f_0^{\mathcal{Q}}(\boldsymbol{x}_0)_{y_0}. \tag{7}$$

In §5.2 we will discuss how to handle confidence intervals of P_A and Q_A. For now, we assume P_A and Q_A are fixed.

Given perturbation vector $\boldsymbol{\delta} \in \mathbb{R}^d$, to test whether smoothed classifier $\widetilde{F}_0^{\mathcal{P}}$ still predicts true label y_0, we only need to check whether the prediction probability $f_0^{\mathcal{P}}(\boldsymbol{x}_0 + \boldsymbol{\delta})_{y_0} > 0.5$. This can be formulated as a constrained optimization problem ($\mathbf{C}$):

$$\begin{aligned}
\underset{f}{\text{minimize}} \quad & \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{P}}[f(\boldsymbol{\epsilon}+\boldsymbol{\delta})] && \text{(8a)}\\
\text{s.t.} \quad & \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{P}}[f(\boldsymbol{\epsilon})] = P_A, \quad \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{Q}}[f(\boldsymbol{\epsilon})] = Q_A, && \text{(8b)}\\
& 0 \le f(\boldsymbol{\epsilon}) \le 1 \quad \forall \boldsymbol{\epsilon} \in \mathbb{R}^d. && \text{(8c)}
\end{aligned}$$

Remark. ($\mathbf{C}$) seeks for the minimum possible $f^{\mathcal{P}}(\boldsymbol{x}_0+\boldsymbol{\delta})_{y_0}$ given Eqn. (7)'s constraint. Concretely, we let f represent whether the base classifier predicts label y_0: $f(\cdot) = \mathbb{I}[F(\cdot + \boldsymbol{x}_0) = y_0]$, and accordingly impose $f \in [0,\, 1]$ in Eqn. (8c). Then, Eqns. (8a) and (8b) unfold $f^{\mathcal{P}}(\boldsymbol{x}_0 + \boldsymbol{\delta})_{y_0}$, $f^{\mathcal{P}}(\boldsymbol{x}_0)_{y_0}$, and $f^{\mathcal{Q}}(\boldsymbol{x}_0)_{y_0}$ respectively and impose Eqn. (7)'s constraint.

We let $\mathbf{C}_{\boldsymbol{\delta}}(P_A,\, Q_A)$ denote the optimal value of Eqn. (8) when feasible. Thus, under norm p, to certify the robustness within radius r, we only need to check whether

$$\forall \boldsymbol{\delta},\, \|\boldsymbol{\delta}\|_p < r \Rightarrow \mathbf{C}_{\boldsymbol{\delta}}(P_A,\, Q_A) > 0.5. \tag{9}$$

This formulation yields the tightest robustness certification given information from $\mathcal{P}$ and $\mathcal{Q}$ under the binary setting. Under the multiclass setting, there are efforts towards tighter certification by using "$>$ maximum over other classes" instead of "> 0.5" in Eqn. (9) (Dvijotham et al., 2020). For saving the sampling cost and also to follow the convention (Cohen et al., 2019; Yang et al., 2020; Jeong & Shin, 2020; Zhai et al., 2020), we mainly consider "> 0.5" for multiclass setting, and extension to the other form is straightforward.

Since our choices of $\mathcal{P}$ and $\mathcal{Q}$ (standard/generalized (truncated) Gaussian) are isotropic and centered around origin, when certifying radius r, for ℓ_2 certification we only need to test $\mathbf{C}_{\boldsymbol{\delta}}(P_A,\, Q_A) > 0.5$ with $\boldsymbol{\delta} = (r,\, 0,\, \ldots,\, 0)^\intercal$; and for ℓ_∞ we only need to divide ℓ_2 radius by $\sqrt{d}$. This trick can also be extended for ℓ_1 case (Zhang et al., 2020a).

Directly solving ($\mathbf{C}$) is challenging. Thus, we construct the Lagrangian dual problem ($\mathbf{D}$):

$$\underset{\lambda_1,\,\lambda_2\in\mathbb{R}}{\text{maximize}} \ \Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[p(\boldsymbol{\epsilon}) < \lambda_1 p(\boldsymbol{\epsilon}+\boldsymbol{\delta}) + \lambda_2 q(\boldsymbol{\epsilon}+\boldsymbol{\delta})] \tag{10a}$$

$$\begin{aligned}
\text{s.t.} \ & \Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[p(\boldsymbol{\epsilon}-\boldsymbol{\delta}) < \lambda_1 p(\boldsymbol{\epsilon}) + \lambda_2 q(\boldsymbol{\epsilon})] = P_A,\\
& \Pr_{\boldsymbol{\epsilon}\sim\mathcal{Q}}[p(\boldsymbol{\epsilon}-\boldsymbol{\delta}) < \lambda_1 p(\boldsymbol{\epsilon}) + \lambda_2 q(\boldsymbol{\epsilon})] = Q_A.
\end{aligned} \tag{10b}$$

In Eqn. (10), $p(\cdot)$ and $q(\cdot)$ are the density functions of distributions $\mathcal{P}$ and $\mathcal{Q}$ respectively. We let $\mathbf{D}_{\boldsymbol{\delta}}(P_A,\, Q_A)$ denote the optimal objective value to Eqn. (10a) when it is feasible.

Theorem 3. *For given $\boldsymbol{\delta} \in \mathbb{R}^d$, P_A, and Q_A, if $\mathbf{C}$ and $\mathbf{D}$ are both feasible, then $\mathbf{C}_{\boldsymbol{\delta}}(P_A,\, Q_A) = \mathbf{D}_{\boldsymbol{\delta}}(P_A,\, Q_A)$.*

The theorem states the strong duality between ($\mathbf{C}$) and ($\mathbf{D}$). We defer the proof to Appendix G.1. The proof is based on min-max inequality and feasibility condition of ($\mathbf{D}$). Intuitively, we can view ($\mathbf{C}$), a functional optimization over f, as a linear programming (LP) problem over infinite number of variables $\{f(\boldsymbol{x}) : \boldsymbol{x} \in \mathbb{R}^d\}$ so that the strong duality holds, which guarantees the tightness of DSRS in the primal space.

5.2. Dealing with Confidence Intervals

It is practically intractable to know the exact P_A and Q_A in Eqn. (7) by only querying the model's prediction for finite times. The common practice is using Monte-Carlo sampling, which gives confidence intervals of P_A and Q_A with a predefined confidence level $1 - \alpha$.

Suppose we have confidence intervals $[\underline{P_A}, \overline{P_A}]$ and $[\underline{Q_A}, \overline{Q_A}]$. To derive a sound certification, we need to certify that for *any* $P_A \in [\underline{P_A}, \overline{P_A}]$ and *any* $Q_A \in [\underline{Q_A}, \overline{Q_A}]$, $\mathbf{C}_{\boldsymbol{\delta}}(P_A, Q_A) > 0.5$. Given the infinite number of possible P_A and Q_A, the brute-force method is intractable. Here, *without* computing $\mathbf{C}_{\boldsymbol{\delta}}$, we show how to solve

$$(P_A, Q_A) = \underset{P'_A \in [\underline{P_A}, \overline{P_A}], Q'_A \in [\underline{Q_A}, \overline{Q_A}]}{\arg\min} \mathbf{C}_{\boldsymbol{\delta}}(P'_A, Q'_A). \quad (11)$$

If solved P_A and Q_A satify $\mathbf{C}_{\boldsymbol{\delta}}(P_A, Q_A) > 0.5$, then for any P_A and Q_A within the confidence intervals, we can certify the robustness against perturbation $\boldsymbol{\delta}$. We observe the following two properties of $\mathbf{C}_{\boldsymbol{\delta}}$.

Proposition 1. *$\mathbf{C}_{\boldsymbol{\delta}}(\cdot, \cdot)$ is convex in the feasible region.*

Proposition 2. *With respect to $x \in [0, 1]$, functions $x \mapsto \min_y \mathbf{C}_{\boldsymbol{\delta}}(x, y)$ and $x \mapsto \arg\min_y \mathbf{C}_{\boldsymbol{\delta}}(x, y)$ are monotonically non-decreasing. Similarly, with respect to $y \in [0, 1]$, functions $y \mapsto \min_x \mathbf{C}_{\boldsymbol{\delta}}(x, y)$ and $y \mapsto \arg\min_x \mathbf{C}_{\boldsymbol{\delta}}(x, y)$ are monotonically non-decreasing.*

These two propositions characterize the landscape of $\mathbf{C}_{\boldsymbol{\delta}}(\cdot, \cdot)$—convex and monotonically non-decreasing along both x and y axes. Thus, desired (P_A, Q_A) (location of minima within the bounded box) lies on the box boundary, and we only need to compute the location of boundary-line-sliced minima and compare it with box constraints to solve Eqn. (11). Formally, we propose an efficient algorithm (Alg. 1, omitted to Appendix E.1) to solve (P_A, Q_A).

Theorem 4. *If Eqn. (11) is feasible, the P_A and Q_A returned by Alg. 1 solve Eqn. (11).*

The above results are proved in Appendix G.2. On a high level, we prove Proposition 1 by definition; we prove Proposition 2 via a reduction to classical Neyman-Pearson-based certification and analysis of this reduced problem; and we prove Theorem 4 based on Propositions 1 and 2 along with exhaustive and nontrivial analyses of all possible cases.

5.3. Solving the Dual Problem

After the smoothed prediction confidences P_A and Q_A are determined from the confidence intervals, now we solve the dual problem $\mathbf{D}_{\boldsymbol{\delta}}(P_A, Q_A)$ as defined in Eqn. (10). We solve the problem based on the following theorem:

Theorem 5 (Numerical Integration for DSRS with Generalized Gaussian Smoothing)**.** *In $\mathbf{D}_{\boldsymbol{\delta}}(P_A, Q_A)$, let $r = \|\boldsymbol{\delta}\|_2$, when $\mathcal{P} = \mathcal{N}^{\mathbf{g}}(k, \sigma)$ and $\mathcal{Q} = \mathcal{N}^{\mathbf{g}}_{\mathsf{trunc}}(k, T, \sigma)$, let $\sigma' := \sqrt{d/(d-2k)}$ and let $\nu := \Gamma\mathrm{CDF}_{d/2-k}(T^2/(2\sigma'^2))$,*

$$R(\lambda_1, \lambda_2) := \Pr_{\boldsymbol{\epsilon} \sim \mathcal{P}}[p(\boldsymbol{\epsilon}) < \lambda_1 p(\boldsymbol{\epsilon} + \boldsymbol{\delta}) + \lambda_2 q(\boldsymbol{\epsilon} + \boldsymbol{\delta})]$$
$$= \begin{cases} \mathbb{E}_{t \sim \Gamma(d/2-k,1)} u_1(t), & \lambda_1 \le 0 \\ \mathbb{E}_{t \sim \Gamma(d/2-k,1)} u_1(t) + u_2(t), & \lambda_1 > 0 \end{cases} \text{ where}$$

$$u_1(t) = \mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{\min\{T^2, 2\sigma'^2 k W(\frac{t}{k} e^{\frac{t}{k}} (\lambda_1 + \nu\lambda_2)^{\frac{1}{k}})\}}{4r\sigma'\sqrt{2t}} - \frac{(\sigma'\sqrt{2t} - r)^2}{4r\sigma'\sqrt{2t}}\right),$$

$$u_2(t) = \max\left\{\mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{2\sigma'^2 k W(\frac{t}{k} e^{\frac{t}{k}} \lambda_1^{\frac{1}{k}}) - (\sigma'\sqrt{2t} - r)^2}{4r\sigma'\sqrt{2t}}\right) - \mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{T^2 - (\sigma'\sqrt{2t} - r)^2}{4r\sigma'\sqrt{2t}}\right), 0\right\},$$

$$P(\lambda_1, \lambda_2) := \Pr_{\boldsymbol{\epsilon} \sim \mathcal{P}}[p(\boldsymbol{\epsilon} - \boldsymbol{\delta}) < \lambda_1 p(\boldsymbol{\epsilon}) + \lambda_2 q(\boldsymbol{\epsilon})]$$
$$= \mathbb{E}_{t \sim \Gamma(d/2-k,1)} \begin{cases} u_3(t, \lambda_1), & t \ge T^2/(2\sigma'^2) \\ u_3(t, \lambda_1 + \nu\lambda_2), & t < T^2/(2\sigma'^2). \end{cases} \text{ where}$$

$$u_3(t, \lambda) = \mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{(r + \sigma'\sqrt{2t})^2}{4r\sigma'\sqrt{2t}} - \frac{2k\sigma'^2 W(\frac{t}{k} e^{\frac{t}{k}} \lambda^{-\frac{1}{k}})}{4r\sigma'\sqrt{2t}}\right),$$

$$Q(\lambda_1, \lambda_2) := \Pr_{\boldsymbol{\epsilon} \sim \mathcal{Q}}[p(\boldsymbol{\epsilon} - \boldsymbol{\delta}) < \lambda_1 p(\boldsymbol{\epsilon}) + \lambda_2 q(\boldsymbol{\epsilon})]$$
$$= \nu \mathbb{E}_{t \sim \Gamma(d/2-k,1)} u_3(t, \lambda_1 + \nu\lambda_2) \cdot \mathbb{I}[t \le T^2/(2\sigma'^2)].$$

In above equations, $\Gamma(d/2-k, 1)$ is gamma distribution and $\Gamma\mathrm{CDF}_{d/2-k}$ is its CDF, $\mathrm{BetaCDF}_{\frac{d-1}{2}}$ is the CDF of distribution $\mathrm{Beta}(\frac{d-1}{2}, \frac{d-1}{2})$, and W is the principal branch of Lambert W function.

When $\mathcal{P}$ is standard Gaussian and $\mathcal{Q}$ is truncated standard Gaussian, we derive similar expressions as detailed in Appendix H.1. We prove Theorem 5 in Appendix G.3. The proof extends the level-set sliced integration and results from (Yang et al., 2020). With the theorem, we can rewrite the dual problem $\mathbf{D}_{\boldsymbol{\delta}}(P_A, Q_A)$ as

$$\max_{\lambda_1, \lambda_2 \in \mathbb{R}} R(\lambda_1, \lambda_2) \,\text{s.t.}\, P(\lambda_1, \lambda_2) = P_A, Q(\lambda_1, \lambda_2) = Q_A, \quad (12)$$

Given concrete λ_1 and λ_2, from the theorem, these function values $P(\lambda_1, \lambda_2)$, $Q(\lambda_1, \lambda_2)$, and $R(\lambda_1, \lambda_2)$ can be easily computed with one-dimensional numerical integration using `SciPy` package.

Now, solving $\mathbf{D}_{\boldsymbol{\delta}}(P_A, Q_A)$ reduces to finding dual variables λ_1 and λ_2 such that $P(\lambda_1, \lambda_2) = P_A$ and $Q(\lambda_1, \lambda_2) = Q_A$. Generally, we find that there is only one unique feasible pair (λ_1, λ_2) for Eqn. (12), so finding out such a pair is sufficient. We prove the uniqueness and discuss how we deal with edge cases where multiple feasible pairs exist in Appendix G.4.

Normally, such solving process is expensive. However, we find a particularly efficient method to solve λ_1 and λ_2 and the algorithm description is in Alg. 3 (in Appendix E.1). At a high level, from Theorem 5, we observe that $Q(\lambda_1, \lambda_2)$ is determined only by the sum $(\lambda_1 + \nu\lambda_2)$ and non-decreasing w.r.t. this sum. Therefore, we apply binary search to find out $(\lambda_1 + \nu\lambda_2)$ that satisfies $Q(\lambda_1, \lambda_2) = Q_A$. Then, we observe that

$$P(\lambda_1, \lambda_2) - \frac{Q(\lambda_1, \lambda_2)}{\nu} = \overbrace{\mathop{\mathbb{E}}_{t \sim \Gamma(d/2-k,1)} u_3(t, \lambda_1) \cdot \mathbb{I}\left[t \ge \frac{T^2}{2\sigma'^2}\right]}^{:= h(\lambda_1)}.$$

Thus, we need to find λ_1 such that $h(\lambda_1) = P_A - Q_A/\nu$. We observe that $h(\lambda_1)$ is non-decreasing w.r.t. λ_1, and we use binary search to solve λ_1. Combining with the value of

$(\lambda_1 + \nu\lambda_2)$, we also obtain λ_2. We lastly leverage numerical integration to compute $R(\lambda_1, \lambda_2)$ following Theorem 5 to solve the dual problem $\mathbf{D}_{\boldsymbol{\delta}}(P_A, Q_A)$.

Practical Certification Soundness. As a practical certification method, we need to guarantee the certification soundness in the presence of numerical error. In DSRS, there are two sources of numerical error: numerical integration error when computing $P(\lambda_1, \lambda_2)$, $Q(\lambda_1, \lambda_2)$, and $R(\lambda_1, \lambda_2)$, and the finite precision of binary search on λ_1 and λ_2. For numerical integration, we notice that typical numerical integration packages such as `scipy` support setting an absolute error threshold Δ and raising warnings when such threshold cannot be reached. We set the absolute threshold $\Delta = 1.5 \times 10^{-8}$, and abstain when the threshold cannot be reached (which never happens in our experimental evaluation). Then, when computing P, Q, and R, suppose the numerical value is v, we use the lower bound $(v - \Delta)$ and upper bound $(v + \Delta)$ in the corresponding context to guarantee the soundness. For the finite precision in binary search, we use the left endpoint or the right endpoint of the final binary search interval to guarantee soundness. For example, we use the left endpoint of λ_1 in R computation, and use the left endpoint of $(\lambda_1 + \nu\lambda_2)$ minus right endpoint of λ_1 to get the lower bound of λ_2 to use in R computation. As a result, we always get an under-estimation of R so the certification is sound. Further discussion is in Appendix E.2.

To this point, we have introduced the DSRS computational method. Complexity and efficiency analysis is omitted to Appendix E.3. Implementation details are in Appendix I.1.

5.4. Extensions

We mainly discussed DSRS computational method for generalized Gaussian $\mathcal{P}$ and truncated generalized Gaussian $\mathcal{Q}$ under ℓ_2 norm. Can we extend it to other settings? Indeed, DSRS is a general framework. In appendices, we show following extensions: (1) DSRS for generalized Gaussian with different variances as $\mathcal{P}$ and $\mathcal{Q}$ (in Appendix H.2); (2) DSRS for other ℓ_p norms (in Appendix H.3); and (3) DSRS that leverages other forms of additional information covering gradient magnitude information (Mohapatra et al., 2020; Levine et al., 2020) (in Appendix L).

6. Experimental Evaluation

In this section, we systematically evaluate DSRS and demonstrate that it achieves tighter certification than the classical Neyman-Pearson-based certification against ℓ_2 perturbations on MNIST, CIFAR-10, and ImageNet. We focus on ℓ_2 certification because additive randomized smoothing is not optimal for other norms (e.g., ℓ_1 (Levine & Feizi, 2021)) or the certification can be directly translated from ℓ_2 certification (e.g., ℓ_∞ (Yang et al., 2020) and semantic transformations (Li et al., 2021)).

6.1. Experimental Setup

Smoothing Distributions. Following Theorem 2, we use generalized Gaussian $\mathcal{N}^{\mathsf{g}}(k, \sigma)$ as smoothing distribution $\mathcal{P}$ where $d/2 - 15 \leq k < d/2$. Specifically, we set k to be $d/2 - 12$ on MNIST, $d/2 - 6$ on CIFAR-10, and $d/2 - 4$ on ImageNet. We use three different σ's: 0.25, 0.50, and 1.00.

In terms of the additional smoothing distribution $\mathcal{Q}$, on MNIST and CIFAR-10, we empirically find that using generalized Gaussian with the same k but different variance yields tighter robustness certification, and therefore we choose σ_g to be 0.2, 0.4, and 0.8 corresponding to $\mathcal{P}$'s σ being 0.25, 0.50, and 1.0, respectively. On ImageNet, the concentration property (see Definition 3) is more pronounced (detail study in Appendix J.1) and thus we use truncated generalized Gaussian $\mathcal{N}^{\mathsf{g}}_{\mathsf{trunc}}(k, T, \sigma)$ as $\mathcal{Q}$. We apply a simple but effective algorithm as explained in Appendix I to determine hyperparameter T in $\mathcal{N}^{\mathsf{g}}_{\mathsf{trunc}}(k, T, \sigma)$.

Models and Training. We consider three commonly-used or state-of-the-art training methods: Gaussian augmentation (Cohen et al., 2019), Consistency (Jeong & Shin, 2020), and SmoothMix (Jeong et al., 2021). We follow the default model architecture on each dataset respectively. We train the models with augmentation noise sampled from the corresponding generalized Gaussian smoothing distribution $\mathcal{P}$. More training details can be found in Appendix I.

Baselines. We consider the Neyman-Pearson-based certification method as the baseline. This certification is widely used and is the tightest given only prediction probability under $\mathcal{P}$ (Cohen et al., 2019; Yang et al., 2020; Jeong & Shin, 2020; Li et al., 2021). We remark that although there are certification methods that leverage more information, to the best of our knowledge, they are not visibly better than the Neyman-Pearson-based method on ℓ_2 certification under practical sampling number (10^5). More comparisons in Appendix J.5 show DSRS is also better than these baselines.

For both baseline and DSRS, following the convention, the certification confidence is $1 - \alpha = 99.9\%$, and we use 10^5 samples for estimating P_A and Q_A. Neyman-Pearson certification does not use the information from additional distribution, and all 10^5 samples are used to estimate the interval of P_A. In DSRS, we use 5×10^4 samples to estimate the interval of P_A with confidence $1 - \frac{\alpha}{2} = 99.95\%$ and the rest 5×10^4 samples for Q_A with the same confidence. By union bound, the whole certification confidence is 99.9%.

Metrics. We uniformly draw 1000 samples from the test set and report *certified robust accuracy* (under each ℓ_2 radius r) that is the fraction of samples that are both correctly classified and have certified robust radii larger than or equal to r. Under each radius r, we report the highest certified robust accuracy among the three variances $\sigma \in \{0.25, 0.50, 1.00\}$ following (Cohen et al., 2019; Salman et al., 2019). We also report evaluation results with ACR metric (Zhai et al., 2020) in Appendix J.3.3.

Table 2. Certified robust accuracy under different radii r with different certification approaches.

Dataset	Training Method	Certification Approach	Certified Accuracy under Radius r											
			0.25	0.50	0.75	1.00	1.25	1.50	1.75	2.00	2.25	2.50	2.75	3.00
MNIST	Gaussian Aug. (Cohen et al., 2019)	Neyman-Pearson	**97.8%**	96.9%	94.6%	88.4%	78.7%	57.6%	41.0%	25.5%	13.6%	6.2%	2.1%	0.9%
		DSRS	**97.8%**	**97.0%**	**95.0%**	**89.8%**	**83.4%**	**61.6%**	**48.4%**	**34.1%**	**21.0%**	**10.6%**	**4.4%**	**1.2%**
	Consistency (Jeong & Shin, 2020)	Neyman-Pearson	**98.4%**	**97.5%**	**96.0%**	92.3%	83.8%	67.5%	49.1%	35.6%	21.7%	10.4%	4.1%	1.9%
		DSRS	**98.4%**	**97.5%**	**96.0%**	**93.5%**	**87.1%**	**71.8%**	**55.8%**	**41.9%**	**31.4%**	**17.8%**	**8.6%**	**2.8%**
	SmoothMix (Jeong et al., 2021)	Neyman-Pearson	**98.6%**	97.6%	96.5%	91.9%	85.1%	73.0%	51.4%	40.2%	31.5%	22.2%	12.2%	4.9%
		DSRS	**98.6%**	**97.7%**	**96.8%**	**93.4%**	**87.5%**	**76.6%**	**54.4%**	**46.2%**	**37.6%**	**29.2%**	**18.5%**	**7.2%**
CIFAR-10	Gaussian Aug. (Cohen et al., 2019)	Neyman-Pearson	56.1%	41.3%	27.7%	18.9%	14.9%	10.2%	7.5%	4.1%	2.0%	0.7%	0.1%	**0.1%**
		DSRS	**57.4%**	**42.7%**	**30.6%**	**20.6%**	**16.1%**	**12.5%**	**8.4%**	**6.4%**	**3.5%**	**1.8%**	**0.7%**	**0.1%**
	Consistency (Jeong & Shin, 2020)	Neyman-Pearson	61.8%	50.9%	38.0%	32.3%	23.8%	19.0%	16.4%	13.8%	11.2%	9.0%	7.1%	5.1%
		DSRS	**62.5%**	**52.5%**	**38.7%**	**35.2%**	**28.1%**	**20.9%**	**17.6%**	**15.3%**	**13.1%**	**10.9%**	**8.9%**	**6.5%**
	SmoothMix (Jeong et al., 2021)	Neyman-Pearson	63.9%	53.3%	40.2%	34.2%	26.7%	20.4%	17.0%	13.9%	10.3%	7.8%	4.9%	2.3%
		DSRS	**64.7%**	**55.5%**	**42.1%**	**35.9%**	**29.4%**	**22.1%**	**18.7%**	**16.1%**	**13.2%**	**10.2%**	**7.1%**	**3.9%**
ImageNet	Guassian Aug. (Cohen et al., 2019)	Neyman-Pearson	57.1%	47.0%	39.3%	33.2%	24.8%	21.4%	17.6%	13.7%	10.2%	7.8%	5.7%	3.6%
		DSRS	**58.4%**	**48.4%**	**41.4%**	**35.3%**	**28.8%**	**23.3%**	**21.3%**	**18.7%**	**14.2%**	**11.0%**	**9.0%**	**5.7%**
	Consistency (Jeong & Shin, 2020)	Neyman-Pearson	59.8%	49.8%	43.3%	36.8%	31.4%	25.6%	22.1%	19.1%	16.1%	14.0%	10.6%	8.5%
		DSRS	**60.4%**	**52.4%**	**44.7%**	**39.3%**	**34.8%**	**28.1%**	**25.4%**	**22.6%**	**19.6%**	**17.4%**	**14.1%**	**10.4%**
	SmoothMix (Jeong et al., 2021)	Neyman-Pearson	46.7%	38.2%	28.8%	24.6%	18.1%	14.2%	11.8%	10.1%	8.9%	7.2%	6.0%	4.6%
		DSRS	**47.4%**	**40.0%**	**30.3%**	**26.8%**	**21.6%**	**15.7%**	**14.0%**	**12.1%**	**9.9%**	**8.4%**	**7.2%**	**5.3%**

6.2. Evaluation Results

We show results in Table 2. The corresponding curves and separated tables for each variance σ are in Appendix J.3.

For *almost all* models and radii r, DSRS yields significantly higher certified accuracy. For example, for Gaussian augmented models, when $r = 2.0$, on MNIST the robust accuracy increases from 25.5% to 34.1% ($+8.6\%$), on CIFAR-10 from 4.1% to 6.4% ($+2.3\%$), and on ImageNet from 13.7% to 18.7% ($+5.0\%$). On average, on MNIST the improvements are around 6% - 9%; on CIFAR-10, the improvements are around 1.5% - 3%; and on ImageNet the improvements are around 2% - 5%. Thus, DSRS can be used in conjunction with different training approaches and provides consistently tighter robustness certification.

The improvements in the robust radius are not as substantial as those in Figure 2(b) (which is around $2\times$). We investigate the reason in Appendix J.1. In summary, the model in Figure 2(b) is trained with standard Gaussian smoothing augmentation and smoothed with generalized Gaussian. The models in this section are trained with generalized Gaussian augmentation. Such training gives higher certified robustness, but in the meantime, gives more advantage to Neyman-Pearson-based certification. This finding implies that there may be a large space for exploring training approaches that favor DSRS certification since all existing training methods are designed for Neyman-Pearson-based certification. Nevertheless, even with these "unsuitable" training methods, DSRS still achieves significantly tighter robustness certification than the baseline.

On the other hand, all the above results are restricted to generalized Gaussian smoothing. We still observe that standard Gaussian smoothing combined with strong training methods (Salman et al., 2019; Jeong & Shin, 2020) and Neyman-Pearson certification (the SOTA setting) yields similar or slightly higher certified robust accuracy than generalized Gaussian smoothing even with DSRS certification. Though DSRS with its suitable generalized Gaussian smoothing does not achieve SOTA certified robustness yet, given the theoretical advantages, we believe that with future tailored training approaches, DSRS with generalized Gaussian smoothing can bring strong certified robustness. More discussion is in Appendix L.3.

Ablation Studies. We present several ablation studies in the appendix. and verify: (1) Effectiveness of our simple heuristic for selecting hyperparameter for $\mathcal{Q}$: We propose a simple heuristic to select the hyperparameter T in smoothing distribution $\mathcal{Q} = \mathcal{N}^{\mathsf{g}}_{\mathsf{trunc}}(k, T, \sigma)$. In Appendix J.2, we propose a gradient-based optimization method to select such $\mathcal{Q}$. We find that our simple heuristic has similar performance compared to the more complex optimization method but is more efficient. (2) Comparison of different types of $\mathcal{Q}$: by choosing different types of $\mathcal{Q}$ distributions (truncated Gaussian or Gaussian with different variance), DSRS has different performance as mentioned in Section 6.1. In Appendix J.4, we investigate the reason. In summary, when concentration property (see Definition 3) is better satisfied, using truncated Gaussian as $\mathcal{Q}$ is better; otherwise, using Gaussian with different variance is better.

7. Conclusion

We propose a general DSRS framework that exploits information based on an additional smoothing distribution to tighten the robustness certification. We theoretically analyze and compare classical Neyman-Pearson and DSRS certification, showing that DSRS has the potential to break the curse of dimensionality of randomized smoothing.

Acknowledgements

We think the anonymous reviewers for their constructive feedback. This work is partially supported by NSF grant No.1910100, NSF CNS No.2046726, C3 AI, and the Alfred P. Sloan Foundation.

References

Albarghouthi, A. Introduction to neural network verification. *arXiv preprint arXiv:2109.10317*, 2021.

Alfarra, M., Bibi, A., Torr, P. H., and Ghanem, B. Data dependent randomized smoothing. *arXiv preprint arXiv:2012.04351*, 2020.

Athalye, A., Carlini, N., and Wagner, D. A. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In Dy, J. G. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pp. 274–283. PMLR, 2018.

Awasthi, P., Jain, H., Rawat, A. S., and Vijayaraghavan, A. Adversarial robustness via robust low rank representations. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, pp. 11391–11403. Curran Associates, Inc., 2020.

Blum, A., Dick, T., Manoj, N., and Zhang, H. Random smoothing might be unable to certify ℓ_∞ robustness for high-dimensional images. *Journal of Machine Learning Research*, 21:211:1–211:21, 2020.

Carlini, N. and Wagner, D. A. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017*, pp. 39–57. IEEE Computer Society, 2017.

Carmon, Y., Raghunathan, A., Schmidt, L., Duchi, J. C., and Liang, P. Unlabeled data improves adversarial robustness. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. B., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 11190–11201. Curran Associates, Inc., 2019.

Chernoff, H. and Scheffe, H. A generalization of the neyman-pearson fundamental lemma. *The Annals of Mathematical Statistics*, 23(4):213–225, 1952.

Chiang, P., Curry, M. J., Abdelkader, A., Kumar, A., Dickerson, J., and Goldstein, T. Detection as regression: Certified object detection with median smoothing. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, pp. 1275–1286. Curran Associates, Inc., 2020.

Cohen, J. M., Rosenfeld, E., and Kolter, J. Z. Certified adversarial robustness via randomized smoothing. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pp. 1310–1320. PMLR, 2019.

Dvijotham, K. D., Hayes, J., Balle, B., Kolter, J. Z., Qin, C., György, A., Xiao, K., Gowal, S., and Kohli, P. A framework for robustness certification of smoothed classifiers using f-divergences. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

Eiras, F., Alfarra, M., Kumar, M. P., Torr, P. H. S., Dokania, P. K., Ghanem, B., and Bibi, A. ANCER: anisotropic certification via sample-wise volume maximization. *arXiv preprint arXiv:2107.04570*, 2021.

Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Xiao, C., Prakash, A., Kohno, T., and Song, D. Robust physical-world attacks on deep learning visual classification. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pp. 1625–1634. Computer Vision Foundation / IEEE Computer Society, 2018.

Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. In Bengio, Y. and LeCun, Y. (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

Gowal, S., Dvijotham, K., Stanforth, R., Bunel, R., Qin, C., Uesato, J., Arandjelovic, R., Mann, T. A., and Kohli, P. Scalable verified training for provably robust image classification. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pp. 4841–4850. IEEE, 2019.

Hayes, J. Extensions and limitations of randomized smoothing for robustness guarantees. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2020, Seattle, WA, USA, June 14-19, 2020*, pp. 3413–3421. Computer Vision Foundation / IEEE, 2020.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016,*

Las Vegas, NV, USA, June 27-30, 2016, pp. 770–778. IEEE Computer Society, 2016.

Jeong, J. and Shin, J. Consistency regularization for certified robustness of smoothed classifiers. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, pp. 10558–10570. Curran Associates, Inc., 2020.

Jeong, J., Park, S., Kim, M., Lee, H., Kim, D., and Shin, J. Smoothmix: Training confidence-calibrated smoothed classifiers for certified robustness. In Ranzato, M., Beygelzimer, A., Dauphin, Y. N., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 30153–30168. Curran Associates, Inc., 2021.

Kumar, A., Levine, A., Feizi, S., and Goldstein, T. Certifying confidence via randomized smoothing. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, pp. 5165–5177. Curran Associates, Inc., 2020a.

Kumar, A., Levine, A., Goldstein, T., and Feizi, S. Curse of dimensionality on randomized smoothing for certifiable robustness. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 5458–5467. PMLR, 2020b.

Lécuyer, M., Atlidakis, V., Geambasu, R., Hsu, D., and Jana, S. Certified robustness to adversarial examples with differential privacy. In *2019 IEEE Symposium on Security and Privacy, SP 2019, San Francisco, CA, USA, May 19-23, 2019*, pp. 656–672. IEEE, 2019.

Lee, G., Yuan, Y., Chang, S., and Jaakkola, T. S. Tight certificates of adversarial robustness for randomly smoothed classifiers. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. B., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 4911–4922. Curran Associates, Inc., 2019.

Levine, A. and Feizi, S. Improved, deterministic smoothing for ℓ_1 certified robustness. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 6254–6264. PMLR, 2021.

Levine, A., Kumar, A., Goldstein, T. A., and Feizi, S. Tight second-order certificates for randomized smoothing. *arXiv preprint arXiv:2010.10549*, 2020.

Li, B., Chen, C., Wang, W., and Carin, L. Certified adversarial robustness with additive noise. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. B., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 9459–9469. Curran Associates, Inc., 2019a.

Li, H., Xu, X., Zhang, X., Yang, S., and Li, B. QEBA: query-efficient boundary-based blackbox attack. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pp. 1218–1227. Computer Vision Foundation / IEEE, 2020a.

Li, L., Zhong, Z., Li, B., and Xie, T. Robustra: Training provable robust neural networks over reference adversarial space. In Kraus, S. (ed.), *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pp. 4711–4717. ijcai.org, 2019b.

Li, L., Xie, T., and Li, B. Sok: Certified robustness for deep neural networks. *arXiv preprint arXiv:2009.04131*, 2020b.

Li, L., Weber, M., Xu, X., Rimanic, L., Kailkhura, B., Xie, T., Zhang, C., and Li, B. TSS: transformation-specific smoothing for robustness certification. In Kim, Y., Kim, J., Vigna, G., and Shi, E. (eds.), *2021 ACM SIGSAC Conference on Computer and Communications Security, CCS 2021, Virtual Event, Republic of Korea, November 15 - 19, 2021*, pp. 535–557. ACM, 2021.

Liu, C., Arnon, T., Lazarus, C., Strong, C. A., Barrett, C. W., and Kochenderfer, M. J. Algorithms for verifying deep neural networks. *Foundations and Trends in Optimization*, 4(3-4):244–404, 2021.

Lozier, D. W. NIST digital library of mathematical functions. *Annals of Mathematics and Artificial Intelligence*, 38(1-3):105–119, 2003.

Mirman, M., Gehr, T., and Vechev, M. T. Differentiable abstract interpretation for provably robust neural networks. In Dy, J. G. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML*

2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018, volume 80 of *Proceedings of Machine Learning Research*, pp. 3575–3583. PMLR, 2018.

Mohapatra, J., Ko, C., Weng, T., Chen, P., Liu, S., and Daniel, L. Higher-order certification for randomized smoothing. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, pp. 4501–4511. Curran Associates, Inc., 2020.

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

Qiu, H., Xiao, C., Yang, L., Yan, X., Lee, H., and Li, B. Semanticadv: Generating adversarial examples via attribute-conditioned image editing. In Vedaldi, A., Bischof, H., Brox, T., and Frahm, J. (eds.), *16th European Conference on Computer Vision, ECCV 2020, Glasgow, UK, August 23-28, 2020, Proceedings, Part XIV*, volume 12359 of *Lecture Notes in Computer Science*, pp. 19–37. Springer, 2020.

Raghunathan, A., Steinhardt, J., and Liang, P. Certified defenses against adversarial examples. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

Salman, H., Li, J., Razenshteyn, I. P., Zhang, P., Zhang, H., Bubeck, S., and Yang, G. Provably robust deep learning via adversarially trained smoothed classifiers. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. B., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 11289–11300. Curran Associates, Inc., 2019.

Schuchardt, J., Wollschläger, T., Bojchevski, A., and Günnemann, S. Localized randomized smoothing for collective robustness certification, 2022. URL https://openreview.net/forum?id=mF122BuAnnW.

Súkeník, P., Kuvshinov, A., and Günnemann, S. Intriguing properties of input-dependent randomized smoothing. *arXiv preprint arXiv:2110.05365*, 2021.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I. J., and Fergus, R. Intriguing properties of neural networks. In Bengio, Y. and LeCun, Y. (eds.), *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

Tramèr, F., Carlini, N., Brendel, W., and Madry, A. On adaptive attacks to adversarial example defenses. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, pp. 1633–1645. Curran Associates, Inc., 2020.

Wang, B., Xu, C., Wang, S., Gan, Z., Cheng, Y., Gao, J., Awadallah, A. H., and Li, B. Adversarial GLUE: A multi-task benchmark for robustness evaluation of language models. In Vanschoren, J. and Yeung, S. (eds.), *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*, volume 1, 2021.

Wong, E. and Kolter, J. Z. Provable defenses against adversarial examples via the convex outer adversarial polytope. In Dy, J. G. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pp. 5283–5292. PMLR, 2018.

Wu, Y., Bojchevski, A., Kuvshinov, A., and Günnemann, S. Completing the picture: Randomized smoothing suffers from the curse of dimensionality for a large family of distributions. In Banerjee, A. and Fukumizu, K. (eds.), *The 24th International Conference on Artificial Intelligence and Statistics, AISTATS 2021, April 13-15, 2021, Virtual Event*, volume 130 of *Proceedings of Machine Learning Research*, pp. 3763–3771. PMLR, 2021.

Xu, K., Shi, Z., Zhang, H., Wang, Y., Chang, K., Huang, M., Kailkhura, B., Lin, X., and Hsieh, C. Automatic perturbation analysis for scalable certified robustness and beyond. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, pp. 1129–1141. Curran Associates, Inc., 2020.

Yang, G., Duan, T., Hu, J. E., Salman, H., Razenshteyn, I. P., and Li, J. Randomized smoothing of all shapes and sizes. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 10693–10705. PMLR, 2020.

Yang, Z., Li, L., Xu, X., Kailkhura, B., Xie, T., and Li, B. On the certified robustness for ensemble models and beyond. In *10th International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022.

Zhai, R., Dan, C., He, D., Zhang, H., Gong, B., Ravikumar, P., Hsieh, C., and Wang, L. MACER: attack-free and scalable robust training via maximizing certified radius. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

Zhang, D., Ye, M., Gong, C., Zhu, Z., and Liu, Q. Black-box certification with randomized smoothing: A functional optimization based framework. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, pp. 2316–2326. Curran Associates, Inc., 2020a.

Zhang, H., Chen, H., Xiao, C., Gowal, S., Stanforth, R., Li, B., Boning, D. S., and Hsieh, C. Towards stable and efficient training of verifiably robust neural networks. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020b.

Zhang, J., Li, L., Li, H., Zhang, X., Yang, S., and Li, B. Progressive-scale boundary blackbox attack via projective gradient estimation. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 12479–12490. PMLR, 2021.

A. Neyman-Pearson Certification

The Neyman-Pearson-based robustness certification is the tightest certification given only prediction probability under $\mathcal{P}$ (Cohen et al., 2019). This certification and its equivalent variants are widely used for randomized smoothing. We use $r_{\mathsf{N-P}}$ to represent the certified radius from the Neyman-Pearson-based method.

If the smoothing distribution $\mathcal{P}$ is standard Gaussian, the following proposition gives the closed-form certified robust radius derived from the Neyman-Pearson lemma (Neyman & Pearson, 1933).

Proposition 3 ((Cohen et al., 2019)). *Under ℓ_2 norm, given input $\boldsymbol{x}_0 \in \mathbb{R}^d$ with true label y_0. Let $\mathcal{P} = \mathcal{N}(\sigma)$ be the smoothing distribution, then Neyman-Pearson-based certification yields certified radius $r_{\mathsf{N-P}} = \sigma\Phi^{-1}(f^{\mathcal{P}}(\boldsymbol{x}_0)_{y_0})$, where Φ^{-1} is the inverse CDF of unit-variance Gaussian.*

For other smoothing distributions, the concretization of the Neyman-Pearson certification method can be found in (Yang et al., 2020).

Remark. In practice, the routine is to use Monte-Carlo sampling to obtain a high-confidence interval of $f^{\mathcal{P}}(\boldsymbol{x}_0)_{y_0}$, which implies a high-confidence certification ($r_{\mathsf{N-P}}$) of robust radius. A tighter radius can be obtained when the runner-up prediction probability is known: $r'_{\mathsf{N-P}} = \frac{\sigma}{2}\left(\Phi^{-1}(f^{\mathcal{P}}(\boldsymbol{x}_0)_{y_0}) - \max_{y\in[C]:y\neq y_0}\Phi^{-1}(f^{\mathcal{P}}(\boldsymbol{x}_0)_y)\right)$. However, due to efficiency concern (for C-way classification the sampling number needs to be more than C times if using $r'_{\mathsf{N-P}}$ for certification instead of $r_{\mathsf{N-P}}$), the standard routine is to only use top-class probability and $r_{\mathsf{N-P}}$ (Cohen et al., 2019, Section 3.2.2). DSRS follows this routine.

B. Illustration of Concentration Assumption on ImageNet

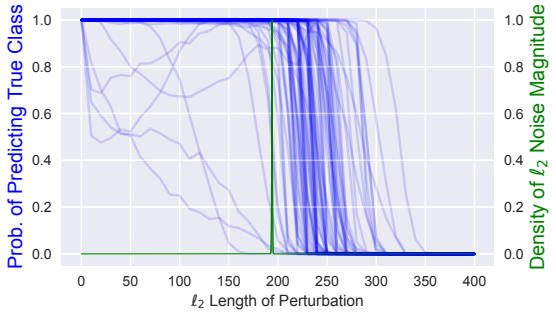


Figure 4. Blue curves: Probability of true-prediction w.r.t. ℓ_2 length of perturbations for a base classifier from (Salman et al., 2019) on ImageNet. Each line corresponds to one of 100 uniformly drawn samples from test set (detailed setup in Appendix J.1). Green curve: Normalized density of ℓ_2 noise magnitude for ImageNet standard Gaussian $\mathcal{N}(\sigma)$ with $\sigma = 0.5$, which highly concentrates on $\sigma\sqrt{d}$. *Thus, for constant P_{con}, $(\sigma, P_{\mathsf{con}})$-concentration can be satisfied for a significant portion of input samples.*

C. Illustration of Unchanged P_A with Increasing d

In the first remark of Theorem 2, we mention that $P_A = f^{\mathcal{P}}(\boldsymbol{x}_0)_{y_0}$ does not grow simultaneously along with the increase of input dimension d. In Figure 5, to illustrate this phenomenon, we plot P_A histograms for $1,000$ test samples from TinyImageNet and ImageNet. Note that TinyImageNet images are downscaled ImageNet images, so the data distribution only differs in the input dimension d. As we can observe, though d varies, the P_A distribution is highly similar, so P_A is roughly unchanged along with the increase of input dimension d.

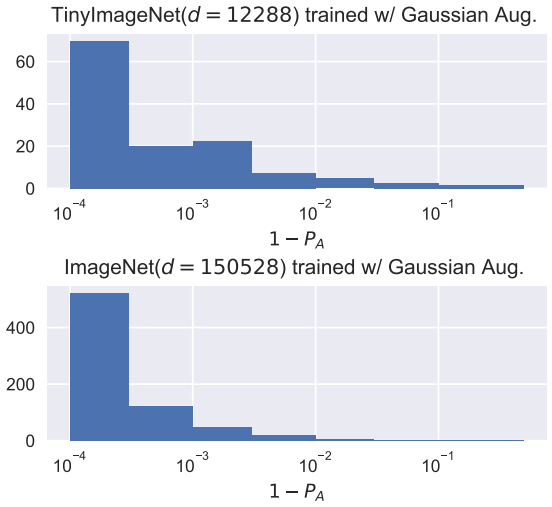


Figure 5. P_A histograms for models trained on TinyImageNet (left) and ImageNet (right) with same $\sigma = 0.50$.

D. DSRS under Relaxed Concentration Assumption

In main text (Section 4), we generalize the concentration assumption by replacing the holding probability 1 in Eqn. (5a) by probability considering sampling confidence. In this appendix, we replace the holding probability in Eqn. (5a) by $\exp(-d^\alpha)$ for $\alpha \in \{0.1, 0.2, 0.3, 0.4, 0, 5\}$.

With this relaxation, we conduct numerical simulations using the same settings as in the main text, and the corresponding results are shown in Figure 6. Note that some solid curves terminate when d is large, which is due to the limitation of floating-point precision in numerical simulations, and we use dashed lines of the same color to plot the projected radius when d is large.

Remark. When the concentration property holds with probability $\exp(-d^\alpha)$ $(0 < \alpha \leq 0.5)$ other than 1, from Figure 6, we observe that $r_{\mathsf{N-P}} d^{\alpha/1.18}$ predicts the certified radius of DSRS well where $r_{\mathsf{N-P}}$ is Neyman-Pearson certified radius. There-

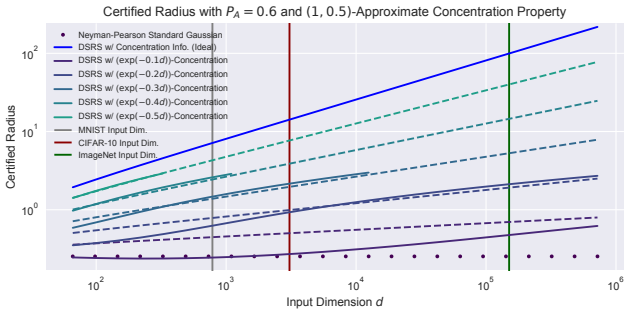


Figure 6. Tendency of DSRS certified robust radius with different input dimensions d under relaxed concentration assumption: when holding probability in Eqn. (5a) is $\exp(-d^\alpha)$ with α from 0.1 to 0.5;. Blue line: DSRS when holding probability in Eqn. (5a) is 1. Dotted line: Neyman-Pearson certification. Other solid lines: DSRS when holding probability in Eqn. (5a) is $\exp(-d^\alpha)$. Other dashed lines: DSRS projected radius by $r^{\text{proj}}_{\text{DSRS}} = r_{\text{N-P}} d^{\alpha/1.18}$. $\alpha \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$. Both x- and y-axes are logarithmic.

fore, although the $\sqrt{d}$ growth rate of ℓ_2 certified radius does not hold, the radius still increases along with the dimension d. Interestingly, along with the increase of dimension d, the vanishing probability $\exp(-d^\alpha)$ still implies the increasing volume of adversarial examples, and smoothed classifier is still certifiably robust with increasing radius reflected by DSRS despite the increasing adversarial volume.

E. Omitted Details of DSRS Computational Method

E.1. Algorithm Description

Algorithm 1 Determining P_A and Q_A from confidence intervals (see §5.2).

Data: Distributions $\mathcal{P}$ and $\mathcal{Q}$; $\boldsymbol{\delta}$; $[\underline{P_A}, \overline{P_A}]$ and $[\underline{Q_A}, \overline{Q_A}]$
Result: P_A and Q_A satisfying Eqn. (11)
1 Compute $\underline{q} \leftarrow \arg\min_y \mathbf{C}_{\boldsymbol{\delta}}(\underline{P_A}, y)$
2 **if** $\underline{q} > \underline{Q_A}$ **then**
3 **return** $(\underline{P_A}, \min\{\underline{q}, \overline{Q_A}\})$
4 **else**
5 Compute $\underline{p} \leftarrow \arg\min_x \mathbf{C}_{\boldsymbol{\delta}}(x, \underline{Q_A})$
6 **return** $(\max\{\min\{\underline{p}, \overline{P_A}\}, \underline{P_A}\}, \underline{Q_A})$

Alg. 1 is a subroutine (Line 7) of Alg. 2. Note that Lines 1 and 5 of Alg. 1 solve the constrained optimization with only one constraint (either one of Eqn. (8b)), reducing to the well-studied and solvable Neyman-Pearson-based certification.

Note that we do not need to evaluate any value of $\mathbf{C}_{\boldsymbol{\delta}}$ in Alg. 1. Although $\underline{q}$ and $\underline{p}$ in the algorithm are "$\arg\min_{x \text{ or } y}$" over $\mathbf{C}_{\boldsymbol{\delta}}$, the free choices of x or y leave $\mathbf{C}_{\boldsymbol{\delta}}$'s constrained optimization with only one constraint and then $\underline{q}$ and $\underline{p}$ can be solved by Neyman-Pearson instead of evaluating $\mathbf{C}_{\boldsymbol{\delta}}$ directly.

Algorithm 2 DSRS computational method.

Data: clean input $\boldsymbol{x}_0$, base classifier F_0; distributions $\mathcal{P}$ and $\mathcal{Q}$; norm type p; confidence level α; numerical integration error bound Δ
Result: Certified radius r
1 Query prediction $y_0 \leftarrow \widetilde{F}_0^{\mathcal{P}}(\boldsymbol{x}_0)$
2 Sample and estimate the intervals of smoothed confidence $[\underline{P_A}, \overline{P_A}]$ under $\mathcal{P}$ and $[\underline{Q_A}, \overline{Q_A}]$ under $\mathcal{Q}$ with confidence $(1-\alpha)$ following (Cohen et al., 2019)
3 Initialize: $r_l \leftarrow 0, r_u \leftarrow r_{\max}$
4 **while** $r_u - r_l >$ eps **do** /* Binary search on radius r */
5 $r_m \leftarrow (r_l + r_u)/2$
6 $\boldsymbol{\delta} \leftarrow (r_m, 0, \ldots, 0)^\intercal$; /* for ℓ_2 certification with ℓ_2 symmetric $\mathcal{P}$ and $\mathcal{Q}$; for ℓ_∞ or ℓ_1, can be adjusted following (Zhang et al., 2020a) */
7 Determine $P_A \in [\underline{P_A}, \overline{P_A}]$ and $Q_A \in [\underline{Q_A}, \overline{Q_A}]$; /* See Section 5.2 and Alg. 1 */
8 $(\lambda_1, \lambda_2) \leftarrow$ DUALBINARYSEARCH(P_A, Q_A) ; /* See Section 5.3 and Alg. 3 */
9 $v \leftarrow R(\lambda_1, \lambda_2) - \Delta$; /* Using Theorem 5 */
10 **if** $v > 0.5$ **then**
11 $r_l \leftarrow r_m$
12 **else**
13 $r_u \leftarrow r_m$
14 **return** r_l

Alg. 2 is the pseudocode of the whole DSRS computational method as introduced in Section 5.

Algorithm 3 DUALBINARYSEARCH for λ_1 and λ_2.

Data: Query access to $P(\cdot,\cdot)$ and $Q(\cdot,\cdot)$; P_A; Q_A; ν; precision parameter ϵ; numerical integration error bound Δ
Result: λ_1 and λ_2 satisfying constraints $P(\lambda_1,\lambda_2) = P_A, Q(\lambda_1,\lambda_2) = Q_A$ (see Eqn. (12))
1 $a^L \leftarrow 0, a^U \leftarrow M$; /* search for $a = \lambda_1 + \nu\lambda_2$, M is a large positive number */
2 **while** $a^U - a^L > \epsilon$ **do**
3 $a^m \leftarrow (a^L + a^U)/2$
4 **if** $Q(a^m, 0) < Q_A$ **then**
5 $a^L \leftarrow a^m$
6 **else**
7 $a^U \leftarrow a^m$
8 **end**
/* Following while-loop enlarges a^L and a^U until $[a^L, a^U]$ covers a^* such that $Q(a^*, 0) = Q_A$ under numerical integration error */
9 **while** $(Q(a^L, 0) + \Delta > Q_A)$ *or* $(Q(a^U, 0) - \Delta < Q_A)$ **do**
10 $t \leftarrow a^U - a^L$
11 $a^L \leftarrow a^L - t/2$
12 $a^U \leftarrow a^U + t/2$
13 **end**
14 $\lambda_1^L \leftarrow 0, \lambda_1^U \leftarrow M$; /* search for λ_1, M is a large positive number */
15 **while** $\lambda_1^U - \lambda_1^L > \epsilon$ **do**
16 $\lambda_1^m \leftarrow (\lambda_1^L + \lambda_1^U)/2$
17 **if** $h(\lambda_1^m) - \Delta < P_A - Q_A/\nu$ **then**
18 $\lambda_1^L \leftarrow \lambda_1^m$
19 **else**
20 $\lambda_1^U \leftarrow \lambda_1^m$
21 **end**
/* Following while-loop enlarges λ_1^L and λ_1^U until $[\lambda_1^L, \lambda_1^U]$ covers λ_1^* such that $h(\lambda_1^*) = P_A - Q_A/\nu$ under numerical integration error */
22 **while** $(h(\lambda_1^L) + \Delta > P_A - Q_A/\nu)$ *or* $(h(\lambda_1^U) - \Delta < P_A - Q_A/\nu)$ **do**
23 $t \leftarrow \lambda_1^U - \lambda_1^L$
24 $\lambda_1^L \leftarrow \lambda_1^L - t/2$
25 $\lambda_1^U \leftarrow \lambda_1^U + t/2$
26 **end**
27 **return** $(\lambda_1^L, (a^L - \lambda_1^U)/\nu)$; /* for soundness, choose the left endpoint of λ_1 and λ_2 range */

Alg. 3 is the dual variable search algorithm described in Section 5.3. From Line 1 to 8, we conduct binary search for quantity $\lambda_1 + \nu\lambda_2$; from Line 14 to 21, we conduct binary search for quantity λ_1. Notice that our binary search interval is initialized to be the non-negative interval. This is because $Q(a^m, 0) = 0$ and $h(\lambda_1^m) = 0$ if a^m and λ_1^m are non-positive observed from Theorem 5.

E.2. Guaranteeing Numerical Soundness

In DSRS computational method, to compute a practically sound robustness guarantee, we take the binary search error and numerical integration error into consideration.

Specifically, in Alg. 3, we return a pair (λ_1', λ_2') whose $R(\lambda_1', \lambda_2')$ lower bounds $R(\lambda_1, \lambda_2)$ where (λ_1, λ_2) is the precise feasible pair. We achieve so by returning $(\lambda_1^L, (a^L - \lambda_1^U)/\nu)$. Specifically, as we can see a^L is an underestimation of actual $(\lambda_1 + \nu\lambda_2)$ in the presence of binary search error and numerical integration error. We bound the numerical integration error by setting absolute error bound $\Delta = 1.5 \times 10^{-8}$ in `scipy.integrad.quad` function. Then, λ_1^L and λ_1^U are underestimation and overestimation of the actual λ_1 in the presence of errors respectively. As a result, $(a^L - \lambda_1^U)/\nu$ is an underestimation of λ_2. Therefore, both λ_1 and λ_2 are underestimated and by the monotonicity of $R(\cdot,\cdot)$, the actual $R(\lambda_1, \lambda_2)$ would be underestimated to guarantee the soundness.

Then, since the evaluation of R involves numerical integration, we compare the lower bound of R: $R(\lambda_1, \lambda_2) - \Delta$ in Alg. 2 (line 9) with 0.5 to determine whether current robustness radius can be certified or not.

E.3. Complexity and Efficiency Analysis

In this appendix, we briefly analyze the computational complexity of DSRS computational method introduced in Section 5. Suppose the binary search precision is ϵ, and each numerical integration costs C time. First, the search of certified robust radius costs $O(\log(\sqrt{d}/\epsilon))$. For each searched radius, we first determine P_A and Q_A by running Neyman-Pearson-based certification, which has cost $O(\log(1/\epsilon)C)$. Then, solving dual variables takes two binary search rounds, which has cost $O(\log(1/\epsilon)C)$. The final one-time integration of $R(\lambda_1, \lambda_2)$ has cost $O(C)$. Thus, overall time complexity is $O(\log(\sqrt{d}/\epsilon)\log(1/\epsilon)C)$, which is the same as classical Neyman-Pearson certification and grows slowly (in logarithmic factor) w.r.t. input dimension d.

In practice, the certification time is on average $5\,\mathrm{s}$ to $10\,\mathrm{s}$ per sample across different datasets. For example, with $\sigma = 0.50$ as the smoothing variance parameter, the certification time, as an overhead over Neyman-Pearson-based certification, is $10.53\,\mathrm{s}$, $4.53\,\mathrm{s}$, and $3.21\,\mathrm{s}$ on MNIST, CIFAR-10, and ImageNet respectively. This overhead is almost negligible compared with the sampling time for estimating P_A and Q_A which is around $200\,\mathrm{s}$ on ImageNet and is the shared cost of all randomized smoothing certification methods. In summary, compared with standard Neyman-Pearson-based certification, the running time of DSRS is roughly the same.

F. Extensions and Proofs in Section 4

In this appendix, we provide formal proofs and theoretical extensions for the results in Section 4.

F.1. Proof of Theorem 1

Proof of Theorem 1. We let D_c denote the decision region of F_0 for class c, i.e., $D_c := \{\boldsymbol{x} : F_0(\boldsymbol{x}) = c\}$. Since $\mathcal{Q}$ is supported on the decision region shifted by $\boldsymbol{x}_0$, $f_0^{\mathcal{Q}}(\boldsymbol{x}_0)_c = 1$. Thus, from $f_0^{\mathcal{Q}}(\boldsymbol{x}_0)_c$, we know $(\mathrm{supp}(\mathcal{Q})+\boldsymbol{x}_0)\backslash S \subseteq D_c$, where S is some set with zero measure under $\mathcal{Q}+\boldsymbol{x}_0$. Since $0 < q(\boldsymbol{x})/p(\boldsymbol{x}) < +\infty$, S also has zero measure under $\mathcal{P}+\boldsymbol{x}_0$. On the other hand, by $0 < q(\boldsymbol{x})/p(\boldsymbol{x}) < +\infty$, we can determine the probability mass of $\mathrm{supp}(\mathcal{Q})$ on $\mathcal{P}$, i.e., $\Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[\boldsymbol{\epsilon} \in \mathrm{supp}(\mathcal{Q})]$. Then, we observe that

$$
\begin{aligned}
&f_0^{\mathcal{P}}(\boldsymbol{x}_0)_c = \Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[F_0(\boldsymbol{x}_0+\boldsymbol{\epsilon}) = c] \\
&= \Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[\boldsymbol{\epsilon} \in \mathbb{R}^d \setminus \mathrm{supp}(\mathcal{Q})] \\
&\qquad \cdot \Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[F_0(\boldsymbol{x}_0+\boldsymbol{\epsilon}) = c \,|\, \boldsymbol{\epsilon} \in \mathbb{R}^d \setminus \mathrm{supp}(\mathcal{Q})] \\
&\quad + \Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[\boldsymbol{\epsilon} \in \mathrm{supp}(\mathcal{Q})] \cdot \Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[F_0(\boldsymbol{x}_0+\boldsymbol{\epsilon}) = c \,|\, \boldsymbol{\epsilon} \in \mathrm{supp}(\mathcal{Q})] \\
&= \Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[\boldsymbol{\epsilon} \in \mathbb{R}^d \setminus \mathrm{supp}(\mathcal{Q})] \\
&\qquad \cdot \Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[F_0(\boldsymbol{x}_0+\boldsymbol{\epsilon}) = c \,|\, \boldsymbol{\epsilon} \in \mathbb{R}^d \setminus \mathrm{supp}(\mathcal{Q})] \\
&\quad + \Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[\boldsymbol{\epsilon} \in \mathrm{supp}(\mathcal{Q})].
\end{aligned}
$$

By the definition of $\mathrm{supp}(\mathcal{Q})$, we observe that $\Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[F_0(\boldsymbol{x}_0+\boldsymbol{\epsilon}) = c \,|\, \boldsymbol{\epsilon} \in \mathbb{R}^d \setminus \mathrm{supp}(\mathcal{Q})] = 0$. As a result, we will find that $f_0^{\mathcal{P}}(\boldsymbol{x}_0)_c = \Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[\boldsymbol{\epsilon} \in \mathrm{supp}(\mathcal{Q})]$. Then the DSRS certification method can know $\big((\mathbb{R}^d \setminus \mathrm{supp}(\mathcal{Q})) + \boldsymbol{x}_0\big) \cap D_c$ has zero measure under $\mathcal{P}+\boldsymbol{x}_0$, i.e., In summary, the certification method can determine that $\mathrm{supp}(\mathcal{Q}) + \boldsymbol{x}_0$ differs from D_c on some set Δ with zero measure under $\mathcal{P}+\boldsymbol{x}_0$. Because $\mathcal{P}$ has positive density everywhere, Δ also has zero measure under $\mathcal{P}+\boldsymbol{x}_0+\boldsymbol{\delta}$ for arbitrary $\boldsymbol{\delta} \in \mathbb{R}^d$. Thus, for arbitrary $\boldsymbol{\delta} \in \mathbb{R}^d$, the certification method can compute out

$$
\begin{aligned}
f_0^{\mathcal{P}}(\boldsymbol{x}_0+\boldsymbol{\delta})_c &= \Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[F_0(\boldsymbol{x}_0+\boldsymbol{\delta}+\boldsymbol{\epsilon}) = c] \\
&= \Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}+\boldsymbol{\delta}}[\boldsymbol{x}_0+\boldsymbol{\epsilon} \in D_c] = \Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}+\boldsymbol{\delta}}[\mathrm{supp}(\mathcal{Q})].
\end{aligned} \tag{13}
$$

Under the binary classification setting, it implies that for any $\boldsymbol{\delta} \in \mathbb{R}^d$, the $f_0^{\mathcal{P}}(\boldsymbol{x}_0+\boldsymbol{\delta})$ can be uniquely determined by DSRS certification method. Since the smoothed classifier's decision at any $\boldsymbol{x}_0+\boldsymbol{\delta}$ is uniquely determiend by $f_0^{\mathcal{P}}(\boldsymbol{x}_0+\boldsymbol{\delta})$ (see Eqn. (2)), the certification method can exactly know $\widetilde{F}_0^{\mathcal{P}}(\boldsymbol{x}+\boldsymbol{\delta})$ for any $\boldsymbol{\delta}$ and thus determine tightest possible certified robust radius r_{tight}. □

F.2. Extending Theorem 1 to Multiclass Setting

For the multiclass setting, we define a variant of DSRS as follows.

Definition 4 ($r_{\mathsf{DSRS}}^{\mathsf{multi}}$)**.** Given $P_A \in [0,1]$ and $Q_A^{\mathsf{multi}} \in \mathbb{R}^{C-1}$,

$$
\begin{aligned}
&r_{\mathsf{DSRS}}^{\mathsf{multi}} := \max r \quad \text{s.t.} \\
&\forall F : \mathbb{R} \to [C], f^{\mathcal{P}}(\boldsymbol{x}_0)_{y_0} = P_A, \\
&f^{\mathcal{Q}_c}(\boldsymbol{x}_0)_c = (Q_A^{\mathsf{multi}})_c, c \in [C-1] \\
&\Rightarrow \forall \boldsymbol{x}, \|\boldsymbol{x}-\boldsymbol{x}_0\|_p < r, \widetilde{F}^{\mathcal{P}}(\boldsymbol{x}) = y_0.
\end{aligned} \tag{14}
$$

In the above definition, $r_{\mathsf{DSRS}}^{\mathsf{multi}}$ is the tightest possible certified radius with prediction probability Q_A^{multi}, where each component of Q_A^{multi}, namely $(Q_A^{\mathsf{multi}})_c$, corresponds to the prediction probability for label c under additional smoothing distribution $\mathcal{Q}_c$. Note that there are $(C-1)$ additional smoothing distributions $\{Q_c\}_{c=1}^{C-1}$ in this generalization for multiclass setting.

With this DSRS generation, the following corollary extends the tightness analysis in Theorem 1 from binary to the multiclass setting.

Corollary 1. *Suppose the original smoothing distribution $\mathcal{P}$ has positive density everywhere, i.e., $p(\cdot) > 0$. For multiclass classification with base classifier F_0, at point $\boldsymbol{x}_0 \in \mathbb{R}^d$, for each class $c \in [C-1]$, let $\mathcal{Q}_c$ be a distribution that satisfies: (1) its support is the decision region of c shifted by $\boldsymbol{x}_0$: $\mathrm{supp}(\mathcal{Q}_c) = \{\boldsymbol{x}-\boldsymbol{x}_0 : F_0(\boldsymbol{x}) = c\}$; (2) for any $\boldsymbol{x} \in \mathrm{supp}(\mathcal{Q})$, $0 < q_c(\boldsymbol{x})/p(\boldsymbol{x}) < +\infty$. Then, plugging $P_A = f_0^{\mathcal{P}}(\boldsymbol{x}_0)_c$ and Q_A^{multi} where $(Q_A^{\mathsf{multi}})_c = f_0^{\mathcal{Q}_c}(\boldsymbol{x}_0)_c$ for $c \in [C-1]$ into Definition 4, we have $r_{\mathsf{DSRS}}^{\mathsf{multi}} = r_{\mathsf{tight}}$ under any ℓ_p ($p \geq 1$).*

Proof of Corollary 1. Similar as the proof of Theorem 1, the certification method can observe that for any $c \in [C-1]$, $f_0^{\mathcal{Q}_c}(\boldsymbol{x}_0)_c = 1$. Thus, the method knows $\mathrm{supp}(\mathcal{Q}_c)+\boldsymbol{x}_0 \approx D_c$ for arbitrary $c \in [C-1]$. Here, the "$\approx$" means that the difference between the two sets has zero measure under $\mathcal{P}+\boldsymbol{x}_0$. Thus, for arbitrary $\boldsymbol{\delta} \in \mathbb{R}^d$ the certification method can precisely compute out $f^{\mathcal{P}}(\boldsymbol{x}_0)_c$ for any $c \in [C-1]$. Since $f^{\mathcal{P}}(\boldsymbol{x}_0) \in \Delta^C$, we also know $f^{\mathcal{P}}(\boldsymbol{x}_0)_C$ and the smoothed classifier's prediction on $\boldsymbol{x}_0+\boldsymbol{\delta}$ can be uniquely determined. Then, following the same argument in Theorem 1's proof, we can determine the tightest possible certified robust radius $r_{\mathsf{DSRS}}^{\mathsf{multi}}$. □

Remark. To achieve the tightest possible certified radius r_{tight}, for binary classification, we only need one extra scalar as the additional information (Q_A), while for multiclass classification, we need $(C-1)$ extra scalars as the additional information ($Q_A^{\mathsf{multi}} \in \mathbb{R}^{C-1}$). Following the convention as discussed in Appendix A, we are interested in tight certification under the binary classification for sampling efficiency concerns. Therefore, we focus on using only one extra scalar (Q_A) to additional information in DSRS. In both Theorem 1 and Corollary 1, we only need finite quantities to achieve tight certification for *any* smoothed classifier. In contrast, other existing work requires infinite quantities to achieve such optimal tightness (Mohapatra et al., 2020, Section 3.1).

F.3. Proof of Theorem 2

The proof of Theorem 2 is a bit complicated, which relies on several propositions and lemmas along with theoretical results in Section 5. At high level, based on the standard Gaussian distribution's property (Proposition F.1), we find $Q_A = 1$ under concentration property (Lemma F.2). With $Q_A = 1$, we derive a lower bound of r_{DSRS} in Lemma F.3. We then use: (1) the concentration of beta distribution $\mathrm{Beta}(\frac{d-1}{2}, \frac{d-1}{2})$ (see Lemma F.4) for large d; (2) the relative concentration of gamma $\Gamma(d/2, 1)$ distribution around mean for large d (see Proposition F.5 and resulting Fact F.7); and (3) the misalignment of gamma distribution $\Gamma(d/2-k, 1)$'s mean and median for small $(d/2-k)$ (see Proposition F.6) to lower bound the quantity in Lemma F.3 and show it is large or equal to 0.5. Then, using the conclusion in Section 5 we conclude that $r_{\mathsf{DSRS}} \geq 0.02\sigma\sqrt{d}$.

Proposition F.1. *If random vector $\boldsymbol{\epsilon} \in \mathbb{R}^d$ follows standard Gaussian distribution $\mathcal{N}(\sigma)$, then*

$$\Pr[\|\boldsymbol{\epsilon}\|_2 \leq T] = \Gamma\mathrm{CDF}_{d/2}\left(\frac{T^2}{2\sigma^2}\right), \tag{15}$$

where $\Gamma\mathrm{CDF}_{d/2}$ is the CDF of gamma distribution $\Gamma(d/2, 1)$.

Proof of Proposition F.1. According to (Lozier, 2003, Eqn. 5.19.4), he volume of a d-dimensional ball, i.e., d-ball, with radius r is $V_d(r) = \frac{\pi^{d/2}}{\Gamma(\frac{d}{2}+1)} r^d$. Thus,

$$\mathrm{Vol}(\{\boldsymbol{\epsilon} : \|\boldsymbol{\epsilon}\|_2 = r\}) = V_d(r)' = \frac{d\pi^{d/2}}{\Gamma(\frac{d}{2}+1)} r^{d-1}. \tag{16}$$

$$\begin{aligned}
&\Pr[\|\boldsymbol{\epsilon}\|_2 \leq T] \\
=&\int_0^T \frac{1}{(2\pi\sigma^2)^{d/2}} \cdot \exp\left(-\frac{r^2}{2\sigma^2}\right) \cdot \mathrm{Vol}(\{\boldsymbol{\epsilon} : \|\boldsymbol{\epsilon}\|_2 = r\})\mathrm{d}r \\
=&\int_0^T \frac{1}{(2\pi\sigma^2)^{d/2}} \cdot \exp\left(-\frac{r^2}{2\sigma^2}\right) \cdot \frac{d\pi^{d/2}}{\Gamma(\frac{d}{2}+1)} r^{d-1}\mathrm{d}r \\
=&\int_0^{T^2} \frac{1}{(2\sigma^2)^{d/2}\Gamma(\frac{d}{2})} \exp\left(-\frac{r}{2\sigma^2}\right) r^{d/2-1}\mathrm{d}r \\
=&\frac{1}{\Gamma(\frac{d}{2})} \int_0^{\frac{T^2}{2\sigma^2}} \exp(-r) r^{d/2-1}\mathrm{d}r = \Gamma\mathrm{CDF}_{d/2}\left(\frac{T^2}{2\sigma^2}\right).
\end{aligned}$$

□

With Proposition F.1, now we can show that $Q_A = 1$ under the condition of Theorem 2 as stated in the following lemma.

Lemma F.2. *Suppose F_0 satisfies $(\sigma, P_{\mathsf{con}})$-concentration property at input point $\boldsymbol{x}_0 \in \mathbb{R}^d$, with additional smoothing distribution $\mathcal{Q} = \mathcal{N}^{\mathsf{g}}_{\mathsf{trunc}}(k, T, \sigma)$ where $T^2 = 2\sigma^2\Gamma\mathrm{CDF}^{-1}_{d/2}(P_{\mathsf{con}})$ and $d/2 - 15 \leq k < d/2$, we have*

$$Q_A = \Pr_{\boldsymbol{\epsilon}\sim\mathcal{Q}}[F_0(\boldsymbol{x}_0 + \boldsymbol{\epsilon}) = y_0] = 1. \tag{17}$$

Proof of Lemma F.2. According to Definition 3, for T' that satisfies

$$\Pr_{\boldsymbol{\epsilon}\sim\mathcal{N}(\sigma)}[\|\boldsymbol{\epsilon}\|_2 \leq T'] = P_{\mathsf{con}} \tag{18}$$

we have

$$\Pr_{\boldsymbol{\epsilon}\sim\mathcal{N}(\sigma)}[F_0(\boldsymbol{x}_0 + \boldsymbol{\epsilon}) = y_0 \,|\, \|\boldsymbol{\epsilon}\|_2 \leq T'] = 1. \tag{19}$$

With Eqn. (18), from Proposition F.1, we have

$$T'^2 = 2\sigma^2\Gamma\mathrm{CDF}^{-1}_{d/2}(P_{\mathsf{con}}). \tag{20}$$

Thus, Eqn. (19) implies

$$\Pr_{\boldsymbol{\epsilon}\sim\mathcal{N}(\sigma)}[F_0(\boldsymbol{x}_0 + \boldsymbol{\epsilon}) = y_0 \,|\, \|\boldsymbol{\epsilon}\|_2 \leq \sqrt{2\sigma^2\Gamma\mathrm{CDF}^{-1}_{d/2}(P_{\mathsf{con}})}] = 1. \tag{21}$$

Notice that $\mathcal{N}(\sigma)$ has finite and positive density anywhere within $\{\boldsymbol{\epsilon} : \|\boldsymbol{\epsilon}\|_2 \leq T'\}$. Thus, $F_0(\boldsymbol{x}_0 + \boldsymbol{\epsilon}) = y_0$ for any $\boldsymbol{\epsilon}$ with $\|\boldsymbol{\epsilon}\|_2 \leq T'$ unless a zero-measure set.

Now, we consider $\mathcal{Q}$. $\mathcal{Q} = \mathcal{N}^{\mathsf{g}}_{\mathsf{trunc}}(k, T, \sigma)$ where $T = T'$, and $\mathcal{N}_{\mathsf{trunc}}$ has finite and positive density anywhere within $\{\boldsymbol{\epsilon} : \|\boldsymbol{\epsilon}\|_2 \leq T'\} \setminus \{\mathbf{0}\}$. Thus,

$$\begin{aligned}
Q_A =& \Pr_{\boldsymbol{\epsilon}\sim\mathcal{N}^{\mathsf{g}}_{\mathsf{trunc}}(k,T,\sigma)}[F_0(\boldsymbol{x}_0 + \boldsymbol{\epsilon}) = y_0] \\
=& \Pr_{\boldsymbol{\epsilon}\sim\mathcal{N}^{\mathsf{g}}(k,\sigma)}[F_0(\boldsymbol{x}_0 + \boldsymbol{\epsilon}) = y_0 \,|\, \|\boldsymbol{\epsilon}\|_2 \leq T] \\
\overset{(*)}{=}& \Pr_{\boldsymbol{\epsilon}\sim\mathcal{N}^{\mathsf{g}}(k,\sigma)}[F_0(\boldsymbol{x}_0 + \boldsymbol{\epsilon}) = y_0 \,|\, \|\boldsymbol{\epsilon}\|_2 \leq T, \boldsymbol{\epsilon} \neq \mathbf{0}] \\
=& 1.
\end{aligned}$$

In the above equations, $(*)$ is because

$$\begin{aligned}
&\Pr_{\boldsymbol{\epsilon}\sim\mathcal{N}^{\mathsf{g}}(k,\sigma)}[\boldsymbol{\epsilon} = \mathbf{0} \,|\, \|\boldsymbol{\epsilon}\|_2 \leq T] \\
\leq& \lim_{r\to 0} \Pr_{\boldsymbol{\epsilon}\sim\mathcal{N}^{\mathsf{g}}(k,\sigma)}[\|\boldsymbol{\epsilon}\|_2 \leq r \,|\, \|\boldsymbol{\epsilon}\|_2 \leq T] \\
=& \lim_{r\to 0} C \int_0^r x^{-2k} \exp\left(-\frac{x^2}{2\sigma'^2}\right) dx^{d-1}\mathrm{d}x \\
=& C \lim_{r\to 0} \int_0^r dx^{d-2k-1} \exp\left(-\frac{x^2}{2\sigma'^2}\right) \mathrm{d}x = 0
\end{aligned}$$

where C is a constant, and the last equality is due to $d/2 > k \Rightarrow d - 2k - 1 \geq 0$. □

With $Q_A = 1$, we can have a lower bound of r_{DSRS} as stated in the following lemma.

Lemma F.3. *Under the same condition as in Lemma F.2, we let* $\mathrm{BetaCDF}_{\frac{d-1}{2}}$ *be the CDF of distributon* $\mathrm{Beta}(\frac{d-1}{2}, \frac{d-1}{2})$*, let*

$$r_0 = \max u \quad \text{s.t.} \mathbb{E}_{t\sim\Gamma(\frac{d}{2}-k)} \mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{T^2-(\sigma'\sqrt{2t}-u)^2}{4u\sigma'\sqrt{2t}}\right) \geq 0.5, \tag{22}$$

and let r_{DSRS} *be the tightest possible certified radius in DSRS under* ℓ_2 *when smoothing distribution* $\mathcal{P} = \mathcal{N}^{\mathfrak{g}}(k,\sigma)$*, then*

$$r_{\mathsf{DSRS}} \geq r_0. \tag{23}$$

Proof of Lemma F.3. The proof shares the same core methodology as DSRS computational method introduced in Section 5. Basically, according to Eqn. (9), for any radius r, let $\boldsymbol{\delta} = (r, 0, \ldots, 0)^\intercal$, if $\mathbf{C}_{\boldsymbol{\delta}}(P_A, Q_A) > 0.5$, then $r_{\mathsf{DSRS}} \geq r$, where $Q_A = 1$ according to Lemma F.2, and by the $(\sigma, P_{\mathsf{con}})$-concentration property

$$P_A \geq \Pr_{\boldsymbol{\epsilon}\sim\mathcal{N}^{\mathfrak{g}}(k,\sigma)}[\|\boldsymbol{\epsilon}\|_2 \leq T]. \tag{24}$$

Therefore, to prove the lemma, we only need to show that when

$$\mathbb{E}_{t\sim\Gamma(\frac{d}{2}-k)} \mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{T^2-(\sigma'\sqrt{2t}-u)^2}{4u\sigma'\sqrt{2t}}\right) \geq 0.5, \tag{25}$$

for any $\boldsymbol{\delta} = (u, 0, \ldots, 0)^\intercal$, $\mathbf{C}_{\boldsymbol{\delta}}(P_A, Q_A) > 0.5$.

By definition (Eqn. (8)),

$$\begin{aligned}
&\mathbf{C}_{\boldsymbol{\delta}}(P_A, Q_A)\\
=&\min_f \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{P}}[f(\boldsymbol{\epsilon}+\boldsymbol{\delta})] \quad \text{s.t.}\\
&\quad \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{P}}[f(\boldsymbol{\epsilon})] = P_A, \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{Q}}[f(\boldsymbol{\epsilon})] = Q_A,\\
&\quad 0 \leq f(\boldsymbol{\epsilon}) \leq 1 \quad \forall \epsilon \in \mathbb{R}^d\\
\geq&\min_f \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{P}}[f(\boldsymbol{\epsilon}+\boldsymbol{\delta})] \quad \text{s.t.}\\
&\quad \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{Q}}[f(\boldsymbol{\epsilon})] = 1,\\
&\quad 0 \leq f(\boldsymbol{\epsilon}) \leq 1 \quad \forall \epsilon \in \mathbb{R}^d\\
=&\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{P}}[f(\boldsymbol{\epsilon}+\boldsymbol{\delta})] \quad \text{where} \quad f(\boldsymbol{\epsilon}) = \begin{cases} 1, & \|\boldsymbol{\epsilon}\|_2 \leq T\\ 0. & \|\boldsymbol{\epsilon}\|_2 > T \end{cases}\\
=:&V.
\end{aligned}$$

We now compute V:

$$\begin{aligned}
V =&\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{P}}[\|\boldsymbol{\epsilon}+\boldsymbol{\delta}\|_2 \leq T]\\
=&\int_{\mathbb{R}^d} p(\boldsymbol{x})\mathbb{I}[\|\boldsymbol{x}+\boldsymbol{\delta}\|_2 \leq T]\mathrm{d}\boldsymbol{x}\\
\overset{(1)}{=}&\int_0^\infty y\mathrm{d}y \int_{\substack{p(\boldsymbol{x})=y\\ \mathbb{I}[\|\boldsymbol{x}+\boldsymbol{\delta}\|_2\leq T}} \frac{\mathrm{d}\boldsymbol{x}}{\|\nabla p(\boldsymbol{x})\|_2}\\
\overset{(2)}{=}&\int_0^\infty y\mathrm{d}y \int_{\substack{p(\boldsymbol{x})=y\\ \mathbb{I}[\|\boldsymbol{x}+\boldsymbol{\delta}\|_2\leq T}} -\frac{\mathrm{d}\boldsymbol{x}}{r_p'(r_p^{-1}(y))}\\
\overset{(3)}{=}&\int_0^\infty y\mathrm{d}y \frac{2\pi^{d/2}}{\Gamma(\frac{d}{2})} r_p^{-1}(y)^{d-1}\cdot\left(-\frac{1}{r_p'(r_p^{-1}(y))}\right)\cdot\\
&\quad \Pr[\|\boldsymbol{x}+\boldsymbol{\delta}\|_2 \leq T \,|\, p(\boldsymbol{x}) = y]\\
\overset{(4)}{=}&\int_0^\infty r_p(t)\mathrm{d}t \frac{2\pi^{d/2}}{\Gamma(\frac{d}{2})} t^{d-1}\Pr[\|\boldsymbol{x}+\boldsymbol{\delta}\|_2 \leq T \,|\, \|\boldsymbol{x}\|_2 = t]\\
\overset{(5)}{=}&\int_0^\infty \frac{1}{(2\sigma'^2)^{d/2-k}\pi^{d/2}}\cdot\frac{\Gamma(d/2)}{\Gamma(d/2-k)} t^{-2k}\exp\left(-\frac{t^2}{2\sigma'^2}\right)\cdot\\
&\quad \frac{2\pi^{d/2}}{\Gamma(\frac{d}{2})} t^{d-1}\Pr[\|\boldsymbol{x}+\boldsymbol{\delta}\|_2 \leq T \,|\, \|\boldsymbol{x}\|_2 = t]\mathrm{d}t\\
=&\frac{1}{\Gamma(\frac{d}{2}-k)}\int_0^\infty t^{d/2-k-1}\exp(-t)\cdot\\
&\quad \Pr[\|\boldsymbol{x}+\boldsymbol{\delta}\|_2 \leq T \,|\, \|\boldsymbol{x}\|_2 = \sigma'\sqrt{2t}]\mathrm{d}t\\
=&\mathbb{E}_{t\sim\Gamma(\frac{d}{2}-k)}\Pr[\|\boldsymbol{x}+\boldsymbol{\delta}\|_2 \leq T \,|\, \|\boldsymbol{x}\|_2 = \sigma'\sqrt{2t}].
\end{aligned}$$

In above equations, (1) follows from level-set sliced integration extended from (Yang et al., 2020) and $p(\boldsymbol{x})$ is the density of distribution $\mathcal{P} = \mathcal{N}^{\mathfrak{g}}(k,\sigma)$ at point $\boldsymbol{x}$; in (2) we define $r_p(\|\boldsymbol{x}\|_2) := p(\boldsymbol{x})$ noting that $\mathcal{P}$ is ℓ_2 symmetric and all $\boldsymbol{x}$ with same ℓ_2 length having the same $p(\boldsymbol{x})$, and we have $\|\nabla p(\boldsymbol{x})\|_2 = -r_p'(r_p^{-1}(y))$ since $y = r_p(\|\boldsymbol{x}\|_2)$ and r_p is monotonically decreasing; (3) uses

$$\begin{aligned}
\mathrm{Vol}(\{\boldsymbol{x} : p(\boldsymbol{x}) = y\}) &= \mathrm{Vol}(\{\boldsymbol{x} : \|\boldsymbol{x}\|_2 = r_p^{-1}(y)\})\\
&= \frac{2\pi^{d/2}}{\Gamma(\frac{d}{2})} r_p^{-1}(y)^{d-1};
\end{aligned} \tag{26}$$

(4) changes the integration variable from y to $t = r_p^{-1}(y)$; and (5) injects the concrete expression of r_p.

Now, we inspect $\Pr[\|\boldsymbol{x}+\boldsymbol{\delta}\|_2 \leq T \,|\, \|\boldsymbol{x}\|_2 = \sigma'\sqrt{2t}]$: When $\|\boldsymbol{x}\|_2 = \sigma'\sqrt{2t}$, $\sum_{i=1}^d x_i^2 = 2t\sigma'^2$. Meanwhile, $\|\boldsymbol{x}+\boldsymbol{\delta}\|_2 \leq T$ means $(x_1+u)^2 + \sum_{i=2}^d x_i^2 \leq T^2$. Thus, when $\|\boldsymbol{x}\|_2 = \sigma'\sqrt{2t}$,

$$\|\boldsymbol{x}+\boldsymbol{\delta}\|_2 \leq T \iff \frac{x_1}{\sigma'\sqrt{2t}} \leq \frac{T^2-u^2-2t\sigma'^2}{2u\sigma'\sqrt{2t}}. \tag{27}$$

According to (Yang et al., 2020, Lemma I.23), for $\boldsymbol{x}$ uniformly sampled from sphere with radius $\sigma'\sqrt{2t}$, the component coordinate $\frac{1+\frac{x_1}{\sigma'\sqrt{2t}}}{2} \sim \mathrm{Beta}(\frac{d-1}{2}, \frac{d-1}{2})$. Thus,

$$\begin{aligned}
&\Pr[\|\boldsymbol{x}+\boldsymbol{\delta}\|_2 \leq T \,|\, \|\boldsymbol{x}\|_2 = \sigma'\sqrt{2t}]\\
=&\mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{1+\frac{T^2-u^2-2t\sigma'^2}{2u\sigma'\sqrt{2t}}}{2}\right)\\
=&\mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{T^2-(\sigma'\sqrt{2t}-u)^2}{4u\sigma'\sqrt{2t}}\right).
\end{aligned} \tag{28}$$

Finally, we get

$$V = \mathbb{E}_{t\sim\Gamma(\frac{d}{2}-k,1)}\mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{T^2-(\sigma'\sqrt{2t}-u)^2}{4u\sigma'\sqrt{2t}}\right). \tag{29}$$

In other words, when

$$\mathbb{E}_{t\sim\Gamma(\frac{d}{2}-k,1)}\mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{T^2-(\sigma'\sqrt{2t}-u)^2}{4u\sigma'\sqrt{2t}}\right) \geq 0.5, \tag{30}$$

we have $V \geq 0.5$, and thus $\mathbf{C}_\delta(P_A, Q_A) \geq V \geq 0.5$, $r_{\mathsf{DSRS}} \geq u$, which concludes the proof. □

We require the following property of BetaCDF.

Lemma F.4. *There exists $d_0 \in \mathbb{N}_+$, for any $d \geq d_0$,*

$$\mathrm{BetaCDF}_{\frac{d-1}{2}}(0.6) \geq 0.999. \tag{31}$$

Proof of Lemma F.4. We let $\mathrm{v} \sim \mathrm{Beta}(\frac{d-1}{2}, \frac{d-2}{2})$, then

$$\mathbb{E}[\mathrm{v}] = 1/2, \tag{32}$$

$$\mathrm{Var}[\mathrm{v}] = \frac{(\frac{d-1}{2})^2}{(\frac{d-1}{2}+\frac{d-1}{2})^2(\frac{d-1}{2}+\frac{d-1}{2}+1)} = \frac{1}{4d}. \tag{33}$$

Now, applying Chebyshev's inequality, we have

$$\Pr[|\mathrm{v}-0.5| \geq 0.1] \leq \frac{1}{0.04d}. \tag{34}$$

Therefore,

$$\begin{aligned}
&\mathrm{BetaCDF}_{\frac{d-1}{2}}(0.6) && (35)\\
=&\Pr[\mathrm{v} < 0.6] && (36)\\
=&1-\Pr[\mathrm{v} \geq 0.6] && (37)\\
\geq&1-\Pr[|\mathrm{v}-0.5| \geq 0.1] \geq 1-\frac{1}{0.04d}. && (38)
\end{aligned}$$

Thus, when $d \geq 25000$, $\mathrm{BetaCDF}_{\frac{d-1}{2}}(0.6) \geq 0.999$. □

We also require the following properties of the gamma distribution.

Proposition F.5. *For any $P_{\mathsf{con}} \in (0,1)$, there exists $d_0 \in \mathbb{N}_+$, for any $d \geq d_0$,*

$$\Gamma\mathrm{CDF}^{-1}_{d/2}(P_{\mathsf{con}}) \geq 0.99 \cdot \frac{d}{2}. \tag{39}$$

Proof of Proposition F.5. We let $\mathrm{v} \sim \Gamma(d/2)$, then

$$\mathbb{E}[\mathrm{v}] = d/2, \tag{40}$$

$$\mathrm{Var}[\mathrm{v}] = d/2. \tag{41}$$

We now apply Chebyshev's inequality and get

$$\begin{aligned}
&\Pr[\mathrm{v} < 0.99 \cdot d/2]\\
\leq&\Pr[|\mathrm{v}-d/2| > 0.01 \cdot d/2]\\
\leq&\frac{20000}{d}.
\end{aligned}$$

Thus, for any $P_{\mathsf{con}} \in (0,1)$, when $d \geq \dfrac{20000}{P_{\mathsf{con}}}$,

$$\Gamma\mathrm{CDF}_{d/2}\left(0.99 \cdot \frac{d}{2}\right) \leq \frac{20000}{d} \leq P_{\mathsf{con}}, \tag{42}$$

i.e., $\Gamma\mathrm{CDF}^{-1}_{d/2}(P_{\mathsf{con}}) \geq 0.99 \cdot \frac{d}{2}$. □

Proposition F.6. *When $d/2-15 \leq k < d/2$, $k, d \in \mathbb{N}_+$,*

$$\Pr_{t\sim\Gamma(d/2-k)}\left[t \leq 0.98\left(\frac{d}{2}-k\right)\right] \geq \frac{0.5}{0.999}. \tag{43}$$

Proof of Proposition F.6. We prove the proposition by enumeration. Notice that $d/2-k \in \{0.5, 1.0, \cdots, 14.5, 15.0\}$, we enumerate $\Gamma\mathrm{CDF}_{d/2-k}(0.98(d/2-k))$ for each $(d/2-k)$ and get the following table.

$\frac{d}{2}-k$	$\Gamma\mathrm{CDF}_{d/2-k}(0.98(d/2-k))$	$\frac{d}{2}-k$	$\Gamma\mathrm{CDF}_{d/2-k}(0.98(d/2-k))$
0.5	0.6778	8.0	0.5245
1.0	0.6247	8.5	0.5224
1.5	0.5990	9.0	0.5204
2.0	0.5831	9.5	0.5186
2.5	0.5718	10.0	0.5168
3.0	0.5632	10.5	0.5152
3.5	0.5564	11.0	0.5136
4.0	0.5507	11.5	0.5121
4.5	0.5459	12.0	0.5107
5.0	0.5418	12.5	0.5093
5.5	0.5381	13.0	0.5080
6.0	0.5349	13.5	0.5068
6.5	0.5319	14.0	0.5056
7.0	0.5292	14.5	0.5044
7.5	0.5268	15.0	0.5033

On the other hand, $\frac{0.5}{0.999} \leq 0.5001$, which concludes the proof. □

Now we are ready to prove the main theorem.

Proof of Theorem 2. According to Lemma F.3, we only need to show that for $u = 0.02\sigma\sqrt{d}$, Eqn. (22) holds. For sufficiently large d, indeed,

$$\begin{aligned}
&\mathbb{E}_{t\sim\Gamma(d/2-k)}\mathrm{BetaCDF}_{(d-1)/2}\left(\frac{T^2-(\sigma'\sqrt{2t}-u)^2}{4u\sigma'\sqrt{2t}}\right)\\
\overset{\text{Lemma F.4}}{\geq}&\ 0.999\mathbb{E}_{t\sim\Gamma(d/2-k)}\mathbb{I}\left[\frac{T^2-(\sigma'\sqrt{2t}-u)^2}{4u\sigma'\sqrt{2t}} \geq 0.6\right]\\
\overset{(*)}{\geq}&\ 0.999\mathbb{E}_{t\sim\Gamma(d/2-k)}\mathbb{I}\left[t \leq 0.98\left(\frac{d}{2}-k\right)\right]\\
\overset{\text{Proposition F.6}}{\geq}&\ 0.999 \cdot \frac{0.5}{0.999} = 0.5.
\end{aligned}$$

Thus, from Lemma F.3 we have $r_{\mathsf{DSRS}} \geq u = 0.02\sqrt{d}$.

The inequality $(*)$ follows from Fact F.7. □

Fact F.7. *Under the condition of Theorem 2, for sufficiently large d,*

$$t \le 0.98\left(\frac{d}{2}-k\right) \Rightarrow \frac{T^2-(\sigma'\sqrt{2t}-u)^2}{4u\sigma'\sqrt{2t}} \ge 0.6. \quad (44)$$

Proof of Fact F.7.

$$\begin{aligned}
&\frac{T^2-(\sigma'\sqrt{2t}-u)^2}{4u\sigma'\sqrt{2t}} \ge 0.6 \\
\iff& T^2-(\sigma'\sqrt{2t}-u)^2 \ge 2.4u\sigma'\sqrt{2t} \\
\overset{x:=\sigma'\sqrt{2t}}{\iff}& T^2-(x-u)^2 \ge 2.4ux \\
\iff& x^2+0.4ux+u^2-T^2 \le 0.
\end{aligned}$$

From Proposition F.5, we have

$$\begin{aligned}
&u^2-T^2 \\
=&0.0004d\sigma^2-2\sigma^2\Gamma\mathrm{CDF}^{-1}_{d/2}(P_{\mathsf{con}}) \quad (45)\\
\le&0.0004d\sigma^2-0.99d\sigma^2<0.
\end{aligned}$$

Thus,

$$\begin{aligned}
&x^2+0.4ux+u^2-T^2 \le 0 \\
\iff& x \le \frac{-0.4u+\sqrt{0.16u^2-4(u^2-T^2)}}{2} \\
&\quad = -0.2u+\sqrt{T^2-0.96u^2} \\
\iff& t \le \frac{(-0.2u+\sqrt{T^2-0.96u^2})^2}{2\sigma'^2}.
\end{aligned}$$

Again, from Proposition F.5,

$$\begin{aligned}
&T^2-0.96u^2 \\
=&2\sigma^2\Gamma\mathrm{CDF}^{-1}_{d/2}(P_{\mathsf{con}})-0.96\times 0.0004d\sigma^2 \quad (46)\\
\ge&(0.99-0.96\times 0.0004)d\sigma^2,
\end{aligned}$$

and therefore

$$\begin{aligned}
&(-0.2u+\sqrt{T^2-0.96u^2})^2 \\
\ge&d\sigma^2\left(-0.004+\sqrt{0.99-0.96\times 0.0004}\right)^2 \approx 0.9816d\sigma^2 \\
\ge&0.98d\sigma^2. \quad (47)
\end{aligned}$$

Then,

$$\begin{aligned}
t \le& \frac{(-0.2u+\sqrt{T^2-0.96u^2})^2}{2\sigma'^2} \\
\Longleftarrow& t \le \frac{0.98d\sigma^2}{2\sigma^2}\cdot\frac{d-2k}{d} \\
\iff& t \le 0.98\left(\frac{d}{2}-k\right).
\end{aligned}$$

□

F.4. Theorem 6

Theorem 6. *Let d be the input dimension and F_0 be the base classifier. For an input point $\boldsymbol{x}_0 \in \mathbb{R}^d$ with true class y_0, suppose F_0 satisfies $(\sigma, P_{\mathsf{con}})$-Concentration property and $\Pr_{\boldsymbol{\epsilon}\sim\mathcal{N}(\sigma)}[F_0(\boldsymbol{x}_0+\boldsymbol{\epsilon})=y_0] = P_{\mathsf{con}}$ where $P_{\mathsf{con}}<1$. The smoothed classifier $\widetilde{F}_0^{\mathcal{P}'}$ is constructed from F_0 and smoothed by generalized Gaussian $\mathcal{P}'=\mathcal{N}^{\mathrm{g}}(k_0,\sigma)$ where k_0 is a constant independent of input dimension d. Then, for any constant $c>0$, there exists d_0, such that when input dimension $d \ge d_0$, **any method cannot certify** ℓ_2 **radius** $c\sqrt{d}$, where $T=\sigma\sqrt{2\Gamma\mathrm{CDF}^{-1}_{d/2}(P_{\mathsf{con}})}$ and $\Gamma\mathrm{CDF}_{d/2}$ is the CDF of gamma distribution $\Gamma(d/2,1)$.*

We defer the proof to Appendix F.5. This theorem suggests that, if we use generalized Gaussian whose k is a constant with respect to input dimension d or use standard Gaussian (whose $k=0$ is a constant) for smoothing, we cannot achieve $\Omega(\sqrt{d})$ certified radius rate from DSRS and any other certification method.

F.5. Proof of Theorem 6

The proof of Theorem 6 is based on three lemmas listed below.

Lemma F.8. *Given $k_0 \in \mathbb{N}$, for any $\epsilon>0$, there exists d_0, such that when $d>d_0$,*

$$\Pr_{\mathrm{t}\sim\Gamma(\frac{d}{2}-k_0,1)}\left[\mathrm{t} \le (1-\epsilon)\left(\frac{d}{2}-k_0\right)\right] \le \frac{0.48}{0.99}. \quad (48)$$

Lemma F.9. *Given $P_{\mathsf{con}} \ge 0$, for any $\epsilon>0$, there exists d_0, such that when $d>d_0$,*

$$T:=\sigma\sqrt{2\Gamma\mathrm{CDF}^{-1}_{d/2}(P_{\mathsf{con}})} \le \sigma\sqrt{(1+\epsilon)d}. \quad (49)$$

Lemma F.10. *For any $\epsilon>0$, there exists d_0, such that when $d>d_0$,*

$$\mathrm{BetaCDF}_{\frac{d-1}{2}}(0.5-\epsilon) \le 0.01. \quad (50)$$

Proofs of these lemmas are based on Chebyshev's inequality.

Proof of Lemma F.8. For $\mathrm{t}\sim\Gamma(d/2-k_0,1)$, we have

$$\mathbb{E}[\mathrm{t}]=d/2-k_0,\ \mathrm{Var}[\mathrm{t}]=d/2-k_0. \quad (51)$$

By Chebyshev's inequality,

$$\begin{aligned}
\Pr\left[\mathrm{t} \le (1-\epsilon)\left(\frac{d}{2}-k_0\right)\right] \le& \Pr\left[|\mathrm{t}-\mathbb{E}[\mathrm{t}]| \ge \epsilon\left(\frac{d}{2}-k_0\right)\right] \\
\le& \frac{1}{\epsilon^2(\frac{d}{2}-k_0)}. \quad (52)
\end{aligned}$$

Picking $d_0=2\left(\frac{0.99}{0.48\epsilon^2}+k_0\right)$ concludes the proof. □

Proof of Lemma F.9. We define random variable $\mathrm{v} \sim \Gamma(d/2, 1)$, so $\mathbb{E}[\mathrm{v}] = d/2, \mathrm{Var}[\mathrm{v}] = d/2$. By Chebyshev's inequality,

$$\begin{aligned}&\Pr[\mathrm{v} \le (1+\epsilon)d/2]\\ \ge& 1 - \Pr[\mathrm{v} \ge (1+\epsilon)d/2]\\ \ge& 1 - \Pr[|\mathrm{v} - \mathbb{E}[\mathrm{v}]| \ge \epsilon d/2]\\ \ge& 1 - \frac{2}{d\epsilon^2}.\end{aligned} \tag{53}$$

Let $d_0 = \frac{2}{\epsilon^2(1-P_{\mathsf{con}})}$. Thus, when $d > d_0$, $\Pr[\mathrm{v} \le (1+\epsilon)d/2] \ge 1 - \frac{2}{d_0\epsilon^2} = P_{\mathsf{con}}$, which implies that $\Gamma\mathrm{CDF}_{d/2}((1+\epsilon)d/2) \ge P_{\mathsf{con}}$ and $\Gamma\mathrm{CDF}^{-1}_{d/2}(P_{\mathsf{con}}) \le (1+\epsilon)d/2$ and concludes the proof. □

Proof of Lemma F.10. We define random variable $\mathrm{v} \sim \mathrm{Beta}(\frac{d-1}{2}, \frac{d-1}{2})$, and we have $\mathbb{E}[\mathrm{v}] = 1/2, \mathrm{Var}[\mathrm{v}] = \frac{1}{4d}$. By Chebyshev's inequality,

$$\Pr[\mathrm{v} \le 0.5 - \epsilon] \le \Pr[|\mathrm{v} - \mathbb{E}[\mathrm{v}]| \ge \epsilon] \le \frac{1}{4d\epsilon^2}. \tag{54}$$

Let $d_0 = \frac{25}{d\epsilon^2}$, when $d > d_0$, $\Pr[\mathrm{v} \le 0.5 - \epsilon] \le 0.01$ and hence $\mathrm{BetaCDF}_{\frac{d-1}{2}}(0.5 - \epsilon) \le 0.01$. □

Now we are ready to prove the main theorem.

Proof of Theorem 6. According to the above three lemmas, we pick d_0, such that when $d > d_0$, the followings hold simultaneously.

$$\Pr_{\mathrm{t}\sim\Gamma\left(\frac{d}{2}-k_0,1\right)}\left[\mathrm{t} \le \left(1 - \frac{c^2}{8\sigma^2}\right)\left(\frac{d}{2} - k_0\right)\right] \le \frac{0.48}{0.99}, \tag{55}$$

$$T = \sigma\sqrt{2\Gamma\mathrm{CDF}^{-1}_{d/2}(P_{\mathsf{con}})} \le \sigma\sqrt{\left(1 + \frac{c^2}{8\sigma^2}\right)d}, \tag{56}$$

$$\mathrm{BetaCDF}_{\frac{d-1}{2}}\left(0.5 - \frac{c}{8\sigma}\right) \le 0.01. \tag{57}$$

We define vector $\boldsymbol{\delta} = (c\sqrt{d}, 0, 0, \ldots, 0)^\intercal$. Since F_0 satisfies $(\sigma, P_{\mathsf{con}})$-concentration property and $\Pr_{\boldsymbol{\epsilon}\sim\mathcal{N}(\sigma)}[F_0(\boldsymbol{x}_0 + \boldsymbol{\epsilon}) = y_0] = P_{\mathsf{con}}$, up to a set of zero measure, the region $\{\boldsymbol{\epsilon} : F_0(\boldsymbol{x}_0 + \boldsymbol{\epsilon}) = y_0\}$ and region $\{\boldsymbol{\epsilon} : \|\boldsymbol{\epsilon}\|_2 \le T\}$ coincide.

We now show that $\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{P}'=\mathcal{N}_{\mathsf{B}}(k_0,\sigma)}[F_0(\boldsymbol{x}_0 + \boldsymbol{\delta} + \boldsymbol{\epsilon}) = y_0] < 0.5$ when $c \le \sigma\sqrt{8/7}$.

$$\begin{aligned}&\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{P}'}[F_0(\boldsymbol{x}_0 + \boldsymbol{\delta} + \boldsymbol{\epsilon}) = y_0]\\ =& \Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}'}[\|\boldsymbol{\delta} + \boldsymbol{\epsilon}\|_2 \le T]\\ =& \mathbb{E}_{\mathrm{t}\sim\Gamma(\frac{d}{2}-k_0,1)}\mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{T^2 - (\sigma'\sqrt{2\mathrm{t}} - c\sqrt{d})^2}{4\sigma'\sqrt{2\mathrm{t}}\cdot c\sqrt{d}}\right)\\ &\qquad\text{(from Eqn. (29))}\\ \le& 0.99\mathbb{E}_{\mathrm{t}\sim\Gamma(\frac{d}{2}-k_0,1)}\mathbb{I}\left[\frac{T^2 - (\sigma'\sqrt{2\mathrm{t}} - c\sqrt{d})^2}{4\sigma'\sqrt{2\mathrm{t}}\cdot c\sqrt{d}} \ge 0.5 - \frac{c}{8\sigma}\right]\\ &+ 0.01 \qquad \text{(by Eqn. (57) and } \mathrm{BetaCDF}_{\cdot}(\cdot) \le 1)\\ =& 0.99\Pr_{\mathrm{t}\sim\Gamma(\frac{d}{2}-k_0,1)}\left[\frac{T^2 - (\sigma'\sqrt{2\mathrm{t}} - c\sqrt{d})^2}{4\sigma'\sqrt{2\mathrm{t}}\cdot c\sqrt{d}} \ge 0.5 - \frac{c}{8\sigma}\right]\\ &+ 0.01.\end{aligned} \tag{58}$$

Since

$$\begin{aligned}&\frac{T^2 - (\sigma'\sqrt{2\mathrm{t}} - c\sqrt{d})^2}{4\sigma'\sqrt{2\mathrm{t}}\cdot c\sqrt{d}} \ge 0.5 - \frac{c}{8\sigma}\\ \iff& T^2 - 2\mathrm{t}\sigma^2\frac{d}{d-2k_0} - dc^2 + \frac{c^2 d}{2}\sqrt{\frac{2\mathrm{t}}{d-2k_0}} \ge 0\end{aligned} \tag{59}$$

$$\overset{\text{Eqn. (56)}}{\Longrightarrow} \left(1 + \frac{c^2}{8\sigma^2}\right)d\sigma^2 - 2\mathrm{t}\sigma^2\frac{d}{d-2k_0} - dc^2 + \frac{c^2 d}{2}\sqrt{\frac{2\mathrm{t}}{d-2k_0}} \ge 0. \tag{60}$$

We now inject $\mathrm{t} = 0$ and $\mathrm{t} = \left(1 - \frac{c^2}{8\sigma^2}\right)\left(\frac{d}{2} - k_0\right)$ to the LHS of Eqn. (60).

- When $\mathrm{t} = 0$,

$$\begin{aligned}\text{LHS of Eqn. (60)} &= \left(1 + \frac{c^2}{8\sigma^2}\right)d\sigma^2 - dc^2\\ &= d\left(\sigma^2 - \frac{7}{8}c^2\right) \ge 0.\end{aligned}$$

- When $\mathrm{t} = \left(1 - \frac{c^2}{8\sigma^2}\right)\left(\frac{d}{2} - k_0\right)$,

$$\begin{aligned}&\text{LHS of Eqn. (60)}\\ =& \left(1 + \frac{c^2}{8\sigma^2}\right)d\sigma^2 - \frac{2d\sigma^2}{d-2k_0}\left(1 - \frac{c^2}{8\sigma^2}\right)\left(\frac{d}{2} - k_0\right)\\ &- dc^2 + \frac{dc^2}{2}\sqrt{1 - \frac{c^2}{8\sigma^2}}\\ =& d\sigma^2 + \frac{dc^2}{8} - d\sigma^2 + \frac{dc^2}{8} - dc^2 + \frac{dc^2}{2}\sqrt{1 - \frac{c^2}{8\sigma^2}}\\ \le& \frac{dc^2}{4} - dc^2 + \frac{dc^2}{2} < 0.\end{aligned}$$

Notice that the LHS of Eqn. (60) is a parabola with negative second-order coefficient. Thus,

$$\text{Eqn. (60)} \Longrightarrow \mathrm{t} \in \left[0, \left(1 - \frac{c^2}{8\sigma^2}\right)\left(\frac{d}{2} - k_0\right)\right] \tag{61}$$

and hence

$$\begin{aligned}&\Pr_{\mathrm{t}\sim\Gamma(\frac{d}{2}-k_0,1)}\left[\frac{T^2-(\sigma'\sqrt{2\mathrm{t}}-c\sqrt{d})^2}{4\sigma'\sqrt{2\mathrm{t}}\cdot c\sqrt{d}}\geq 0.5-\frac{c}{8\sigma}\right]\\ \leq&\Pr_{\mathrm{t}\sim\Gamma(\frac{d}{2}-k_0,1)}\left[\mathrm{t}\leq\left(1-\frac{c^2}{8\sigma^2}\right)\left(\frac{d}{2}-k_0\right)\right]\\ \leq&\frac{0.48}{0.99}. \qquad \text{(by Eqn. (55))}\end{aligned} \tag{62}$$

Plugging this inequality to Eqn. (58), we get

$$\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{P}'}[F_0(\boldsymbol{x}_0+\boldsymbol{\delta}+\boldsymbol{\epsilon})=y_0]\leq 0.99\cdot\frac{0.48}{0.99}+0.01=0.49. \tag{63}$$

As a result, when $c\leq\sigma\sqrt{8/7}$, the smoothed classifier $\widetilde{F}_0^{\mathcal{P}'}$ is not robust given the perturbation $\boldsymbol{\delta}=(c\sqrt{d},0,0,\cdots,0)^\intercal$, since there may exist another $y'\neq y_0$ with $\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{P}'}[F_0(\boldsymbol{x}_0+\boldsymbol{\delta}+\boldsymbol{\epsilon})=y']\geq 0.51$ so $\widetilde{F}_0^{\mathcal{P}'}(\boldsymbol{x}_0+\boldsymbol{\delta})=y'\neq y_0$.

When $c>\sigma\sqrt{8/7}$, within the ℓ_2 radius ball $c\sqrt{d}$, there exists perturbation vector $\boldsymbol{\delta}=(c'\sqrt{d},0,0,\cdots,0)^\intercal$ fooling smoothed classifier $\widetilde{F}_0^{\mathcal{P}'}$ where $c'=\sigma\sqrt{8/7}$. Hence, for any $c>0$, there exists a perturbation within ℓ_2 ball with radius $c\sqrt{d}$, such that smoothed classifier $\widetilde{F}_0^{\mathcal{P}'}$ can be fooled, and then any robustness certification method cannot certify ℓ_2 radius $c\sqrt{d}$ since the smoothed classifier itself is not robust. □

G. Proofs of DSRS Computational Method

G.1. Proof of Strong Duality (Theorem 3)

Proof of Theorem 3. We write down the Lagrangian dual function of Eqn. (8a):

$$\begin{aligned}\Lambda(f,\,\lambda_1,\,\lambda_2):=&\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{P}}[f(\boldsymbol{\delta}+\boldsymbol{\epsilon})]-\lambda_1\left(\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{P}}[f(\boldsymbol{\epsilon})]-P_A\right)\\ &-\lambda_2\left(\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{Q}}[f(\boldsymbol{\epsilon})]-Q_A\right).\end{aligned} \tag{64}$$

Then, from $\mathbf{C}$'s expression (Eqn. (8)), we have

$$\begin{aligned}&\mathbf{C}_{\boldsymbol{\delta}}(P_A,Q_A)\\ =&\min_f\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{P}}[f(\boldsymbol{\epsilon}+\boldsymbol{\delta})]\,\mathrm{s.t.}\,0\leq f(\boldsymbol{\epsilon})\leq 1\,\forall\boldsymbol{\epsilon}\in\mathbb{R}^d,\\ &\qquad\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{P}}[f(\boldsymbol{\epsilon})]=P_A,\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{Q}}[f(\boldsymbol{\epsilon})]=Q_A\\ =&\min_{f:\mathbb{R}^d\to[0,1]}\max_{\lambda_1,\lambda_2\in\mathbb{R}}\Lambda(f,\,\lambda_1,\,\lambda_2)\\ \overset{(i)}{\geq}&\max_{\lambda_1,\lambda_2\in\mathbb{R}}\min_{f:\mathbb{R}^d\to[0,1]}\Lambda(f,\,\lambda_1,\,\lambda_2)\\ \overset{(ii)}{=}&\max_{\lambda_1,\lambda_2\in\mathbb{R}}\Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[p(\boldsymbol{\epsilon})<\lambda_1p(\boldsymbol{\epsilon}+\boldsymbol{\delta})+\lambda_2q(\boldsymbol{\epsilon}+\boldsymbol{\delta})]\\ &\qquad-\lambda_1\Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[p(\boldsymbol{\epsilon}-\boldsymbol{\delta})<\lambda_1p(\boldsymbol{\epsilon})+\lambda_2q(\boldsymbol{\epsilon})]\\ &\qquad-\lambda_2\Pr_{\boldsymbol{\epsilon}\sim\mathcal{Q}}[p(\boldsymbol{\epsilon}-\boldsymbol{\delta})<\lambda_1p(\boldsymbol{\epsilon})+\lambda_2q(\boldsymbol{\epsilon})]\\ &\qquad+\lambda_1P_A+\lambda_2Q_A.\end{aligned} \tag{65}$$

In the above equation, (i) is from the min-max inequality. For completeness, we provide the proof as such: Define $g:\mathbb{R}^2\to\mathbb{R}$ such that $g(\lambda_1,\lambda_2):=\min_{f:\mathbb{R}^d\to[0,1]}\Lambda(f,\,\lambda_1,\lambda_2)$. As a result, for any $\lambda_1,\lambda_2\in\mathbb{R}$ and any $f:\mathbb{R}^d\to[0,1]$, $g(\lambda_1,\lambda_2)\leq\Lambda(f,\lambda_1,\lambda_2)$. So for any $f:\mathbb{R}^d\to[0,1]$, $\max_{\lambda_1,\lambda_2\in\mathbb{R}}g(\lambda_1,\lambda_2)\leq\max_{\lambda_1,\lambda_2\in\mathbb{R}}\Lambda(f,\lambda_1,\lambda_2)$, which implies

$$\max_{\lambda_1,\lambda_2\in\mathbb{R}}g(\lambda_1,\lambda_2)\leq\min_{f:\mathbb{R}\to[0,1]}\max_{\lambda_1,\lambda_2\in\mathbb{R}}\Lambda(f,\lambda_1,\lambda_2), \tag{66}$$

where LHS is the RHS of (i) and RHS is the LHS of (i).

In above equation, (ii) comes from a closed-form solution of f for $\Lambda(f,\lambda_1,\lambda_2)$ given $(\lambda_1,\lambda_2)\in\mathbb{R}^2$. Notice that we can rewrite $\Lambda(f,\lambda_1,\lambda_2)$ as an integral over $\mathbb{R}^d$:

$$\begin{aligned}&\Lambda(f,\,\lambda_1,\,\lambda_2)\\ =&\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{P}}[f(\boldsymbol{\delta}+\boldsymbol{\epsilon})]-\lambda_1\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{P}}[f(\boldsymbol{\epsilon})]-\lambda_2\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{Q}}[f(\boldsymbol{\epsilon})]\\ &\qquad+\lambda_1P_A+\lambda_2Q_A\\ =&\int_{\mathbb{R}^d}f(\boldsymbol{x})\cdot(p(\boldsymbol{x}-\boldsymbol{\delta})-\lambda_1p(\boldsymbol{x})-\lambda_2q(\boldsymbol{x}))\,\mathrm{d}\boldsymbol{x}\\ &\qquad+\lambda_1P_A+\lambda_2Q_A.\end{aligned} \tag{67}$$

We would like to minimize over $f:\mathbb{R}^d\to[0,1]$ in Eqn. (67) and simple greedy solution reveals that we should choose

$$f(\boldsymbol{x})=\begin{cases}1, & p(\boldsymbol{x}-\boldsymbol{\delta})-\lambda_1p(\boldsymbol{x})-\lambda_2q(\boldsymbol{x})<0\\ 0. & p(\boldsymbol{x}-\boldsymbol{\delta})-\lambda_1p(\boldsymbol{x})-\lambda_2q(\boldsymbol{x})\geq 0\end{cases} \tag{68}$$

We inject this f into Eqn. (67) and get

$$\begin{aligned}&\min_{f:\mathbb{R}^d\to[0,1]}\Lambda(f,\,\lambda_1,\,\lambda_2)\\ =&\Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[p(\boldsymbol{\epsilon})<\lambda_1p(\boldsymbol{\epsilon}+\boldsymbol{\delta})+\lambda_2q(\boldsymbol{\epsilon}+\boldsymbol{\delta})]\\ &\qquad+\lambda_1\left(P_A-\Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[p(\boldsymbol{\epsilon}-\boldsymbol{\delta})<\lambda_1p(\boldsymbol{\epsilon})+\lambda_2q(\boldsymbol{\epsilon})]\right)\\ &\qquad+\lambda_2\left(Q_A-\Pr_{\boldsymbol{\epsilon}\sim\mathcal{Q}}[p(\boldsymbol{\epsilon}-\boldsymbol{\delta})<\lambda_1p(\boldsymbol{\epsilon})+\lambda_2q(\boldsymbol{\epsilon})]\right).\end{aligned} \tag{69}$$

Hence (ii) holds.

On the other hand, we know that $\mathbf{D}_{\boldsymbol{\delta}}(P_A,Q_A)$ (defined by Eqn. (10)) is feasible by theorem statement. Denote $(\lambda_1^*,\lambda_2^*)\in\mathbb{R}^2$ to a feasible solution to $\mathbf{D}_{\boldsymbol{\delta}}(P_A,Q_A)$ and d^* to the objective value, then from the constraints of $(\mathbf{D})$ we know

$$\begin{aligned}\Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[p(\boldsymbol{\epsilon}-\boldsymbol{\delta})<\lambda_1^*p(\boldsymbol{\epsilon})+\lambda_2^*q(\boldsymbol{\epsilon})]=P_A,\\ \Pr_{\boldsymbol{\epsilon}\sim\mathcal{Q}}[p(\boldsymbol{\epsilon}-\boldsymbol{\delta})<\lambda_1^*p(\boldsymbol{\epsilon})+\lambda_2^*q(\boldsymbol{\epsilon})]=Q_A,\\ \Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[p(\boldsymbol{\epsilon})<\lambda_1^*p(\boldsymbol{\epsilon}+\boldsymbol{\delta})+\lambda_2^*q(\boldsymbol{\epsilon}+\boldsymbol{\delta})]=d^*.\end{aligned} \tag{70}$$

Plugging in these equalities into Eqn. (65), we have

$$\mathbf{C}_{\boldsymbol{\delta}}(P_A,Q_A)\geq d^*-\lambda_1P_A-\lambda_2Q_A+\lambda_1P_A+\lambda_2Q_A=d^*. \tag{71}$$

At the same time, we define function $f^* : \mathbb{R}^d \to [0, 1]$ such that

$$f^*(\boldsymbol{x}) = \mathbb{I}[p(\boldsymbol{x} - \boldsymbol{\delta}) - \lambda_1^* p(\boldsymbol{x}) - \lambda_2^* q(\boldsymbol{x}) < 0]. \quad (72)$$

From Eqn. (70), f^* satisfies the constraints of ($\mathbf{C}$) (Eqns. (8b) and (8c)). Since ($\mathbf{C}$) minimizes over all possible functions $f : \mathbb{R}^d \to [0, 1]$, we have

$$\mathbf{C}_{\boldsymbol{\delta}}(P_A, Q_A) \leq \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{P}}[f^*(\boldsymbol{\epsilon} + \boldsymbol{\delta})] = d^*. \quad (73)$$

Combining Eqns. (71) and (73), we get $\mathbf{C}_{\boldsymbol{\delta}}(P_A, Q_A) = d^*$ and hence the strong duality holds. □

G.2. Proofs of Propositions 1 and 2 and Theorem 4

Proof of Proposition 1. Suppose f_1 is the optimal solution to $\mathbf{C}_{\boldsymbol{\delta}}(P_A^1, Q_A^1)$ and f_2 is the optimal solution to $\mathbf{C}_{\boldsymbol{\delta}}(P_A^2, Q_A^2)$. Due to the linearity of expectation, $(f_1 + f_2)/2$ satisfies all the constraints of Eqn. (8) for $P_A = (P_A^1 + P_A^2)/2$ and $Q_A = (Q_A^1 + Q_A^2)/2$, i.e., $(f_1 + f_2)/2$ is feasible for $P_A = (P_A^1 + P_A^2)/2$ and $Q_A = (Q_A^1 + Q_A^2)/2$ with objective value $\left(\mathbf{C}_{\boldsymbol{\delta}}(P_A^1, Q_A^1) + \mathbf{C}_{\boldsymbol{\delta}}(P_A^2, Q_A^2)\right)/2$. Thus, we have

$$\mathbf{C}_{\boldsymbol{\delta}}\left(\frac{P_A^1 + P_A^2}{2}, \frac{Q_A^1 + Q_A^2}{2}\right) \leq \frac{1}{2}\left(\mathbf{C}_{\boldsymbol{\delta}}(P_A^1, Q_A^1) + \mathbf{C}_{\boldsymbol{\delta}}(P_A^2, Q_A^2)\right) \quad (74)$$

since $\mathbf{C}$ is a minimization problem. By definition, $\mathbf{C}_{\boldsymbol{\delta}}(P_A, Q_A)$ is convex. □

Remark. Since $\mathbf{C}_{\boldsymbol{\delta}}(P_A, Q_A)$ is defined on a compact $\mathbb{R}^2$ subspace, the convexity implies continuity. The continuity property is used in the following proof of Theorem 4.

Proof of Proposition 2. Here, we only prove the monotonicity for functions $x \mapsto \min_y \mathbf{C}_{\boldsymbol{\delta}}(x, y)$ and $x \mapsto \arg\min_y \mathbf{C}_{\boldsymbol{\delta}}(x, y)$. The same statement for $y \mapsto \min_x \mathbf{C}_{\boldsymbol{\delta}}(x, y)$ and $y \mapsto \min_x \mathbf{C}_{\boldsymbol{\delta}}(x, y)$ is then straightforward due to the symmetry.

For simplification, we define $\mathbf{C}'_{\boldsymbol{\delta}} : x \mapsto \min_y \mathbf{C}_{\boldsymbol{\delta}}(x, y)$ and let $\tilde{\mathbf{C}}_{\boldsymbol{\delta}} : x \mapsto \arg\min_y \mathbf{C}_{\boldsymbol{\delta}}(x, y)$. We notice that both functions can be exactly mapped to the constrained optimization problem ($\mathbf{C}'$) which removes the second constraint in Eqn. (8b) in ($\mathbf{C}$):

$$\begin{aligned} \underset{f}{\text{minimize}} \quad & \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{P}}[f(\boldsymbol{\delta} + \boldsymbol{\epsilon})] && (75a)\\ \text{s.t.} \quad & \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{P}}[f(\boldsymbol{\epsilon})] = x, && (75b)\\ & 0 \leq f(\boldsymbol{\epsilon}) \leq 1 \quad \forall \boldsymbol{\epsilon} \sim \mathbb{R}^d. && (75c) \end{aligned}$$

$\mathbf{C}'_{\boldsymbol{\delta}}(x)$ is the optimal objective to ($\mathbf{C}'$) and $\bar{\mathbf{C}}_{\boldsymbol{\delta}}(x)$ is $\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{Q}}[f^*(\boldsymbol{\epsilon})]$ where f^* is the optimal solution.

Either based on Neyman-Pearson lemma [1933] or strong duality, ($\mathbf{C}'$) is equivalent to ($\mathbf{D}'$) defined as such:

$$\begin{aligned} & \Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[p(\boldsymbol{\epsilon}) < \lambda p(\boldsymbol{\epsilon} + \boldsymbol{\delta})] && (76a)\\ \text{s.t.} \quad & \Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[p(\boldsymbol{\epsilon} - \boldsymbol{\delta}) < \lambda p(\boldsymbol{\epsilon})] = x. && (76b) \end{aligned}$$

For a given x, we only need to find λ satisfying Eqn. (76b). Then,

$$\mathbf{C}'_{\boldsymbol{\delta}}(x) = \Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[p(\boldsymbol{\epsilon}) < \lambda p(\boldsymbol{\epsilon} + \boldsymbol{\delta})], \quad (77)$$

$$\bar{\mathbf{C}}_{\boldsymbol{\delta}}(x) = \Pr_{\boldsymbol{\epsilon}\sim\mathcal{Q}}[p(\boldsymbol{\epsilon} - \boldsymbol{\delta}) < \lambda p(\boldsymbol{\epsilon})]. \quad (78)$$

Now the monotonicity (what we would like to prove) is apparent. For $x_1 < x_2$, from Eqn. (76b), we have $\lambda_1 < \lambda_2$, since the probability density function p is non-negative. Thus, we inject λ_1 and λ_2 into Eqn. (77) and Eqn. (78), and yield

$$\mathbf{C}'_{\boldsymbol{\delta}}(x_1) \leq \mathbf{C}'_{\boldsymbol{\delta}}(x_2),\ \bar{\mathbf{C}}_{\boldsymbol{\delta}}(x_1) \leq \bar{\mathbf{C}}_{\boldsymbol{\delta}}(x_2), \quad (79)$$

which concludes the proof. □

Proof of Theorem 4. We discuss the cases according to the branching statement in the algorithm (Alg. 1).

If $\underline{q} > \underline{Q_A}$,

- if $\underline{q} \leq \overline{Q_A}$, by definition we have $\mathbf{C}_{\boldsymbol{\delta}}(\underline{P_A}, \underline{q}) \leq \mathbf{C}_{\boldsymbol{\delta}}(\underline{P_A}, y)$ for arbitrary y. According to Proposition 2, we also have $\mathbf{C}_{\boldsymbol{\delta}}(\underline{P_A}, \underline{q}) \leq \mathbf{C}_{\boldsymbol{\delta}}(x, y)$ for arbitrary $x \geq \underline{P_A}$ and arbitrary y. Given that $(\underline{P_A}, \underline{q}) \in [\underline{P_A}, \overline{P_A}] \times [\underline{Q_A}, \overline{Q_A}]$, $(\underline{P_A}, \underline{q})$ solves Eqn. (11);
- if $\underline{q} > \overline{Q_A}$, by convexity, $\mathbf{C}_{\boldsymbol{\delta}}(\underline{P_A}, \overline{Q_A}) \leq \mathbf{C}_{\boldsymbol{\delta}}(\underline{P_A}, y)$ for $y \in [\underline{Q_A}, \overline{Q_A}]$.

  We further show that $\mathbf{C}_{\boldsymbol{\delta}}(\underline{P_A}, \overline{Q_A}) \leq \mathbf{C}_{\boldsymbol{\delta}}(x, \overline{Q_A})$ for $x \in [\underline{P_A}, \overline{P_A}]$: assume that this is not true, by Proposition 1, the function $y \mapsto \arg\min_x \mathbf{C}_{\boldsymbol{\delta}}(x, y)$ has function value larger than $\underline{P_A}$ at $y = \overline{Q_A}$. Since $\mathbf{C}_{\boldsymbol{\delta}}(0, 0) = 0$ is the global minimum of $\mathbf{C}_{\boldsymbol{\delta}}$, the function value at $y = 0$ is $x = 0$. By Proposition 2, there exists $y_0 \in [0, \overline{Q_A}]$ such that $\underline{P_A} = \arg\min_x \mathbf{C}_{\boldsymbol{\delta}}(x, y_0)$. Then, we get

$$\begin{aligned} & \mathbf{C}_{\boldsymbol{\delta}}(\underline{P_A}, y_0)\\ & \overset{(i.)}{\leq} \mathbf{C}_{\boldsymbol{\delta}}(\arg\min_x \mathbf{C}_{\boldsymbol{\delta}}(x, \overline{Q_A}), \overline{Q_A})\\ & \overset{(ii.)}{\leq} \mathbf{C}_{\boldsymbol{\delta}}(\underline{P_A}, \overline{Q_A}), \end{aligned} \quad (80)$$

  where $(i.)$ follows from Proposition 2 for $y \mapsto \arg\min_x \mathbf{C}_{\boldsymbol{\delta}}(x, y)$; $(ii.)$ is implied in the meaning of $\arg\min_x \mathbf{C}_{\boldsymbol{\delta}}(x, \overline{Q_A})$. Since $y_0 \in [0, \overline{Q_A}]$, Eqn. (80)

implies that $\underline{q}$ should be in $[0,\ \overline{Q_A}]$ as well, which violates the branching condition. Thus, $\mathbf{C}_{\boldsymbol{\delta}}(\underline{P_A},\ \overline{Q_A}) \leq \mathbf{C}_{\boldsymbol{\delta}}(x,\ \overline{Q_A})$ for $x \in [\underline{P_A},\ \overline{P_A}]$.

Using Proposition 2 for function $x \mapsto \arg\min_y \mathbf{C}_{\boldsymbol{\delta}}(x,\ y)$ in interval $[\underline{P_A},\ \overline{P_A}]$ together with Proposition 1, we get $\mathbf{C}_{\boldsymbol{\delta}}(x,\ \overline{Q_A}) \leq \mathbf{C}_{\boldsymbol{\delta}}(x,\ y)$ for $x \in [\underline{P_A},\ \overline{P_A}]$ and $y \in [\underline{Q_A},\ \overline{Q_A}]$. Thus, $(\underline{P_A},\ \overline{Q_A})$ solves Eqn. (11).

If $\underline{q} \leq \underline{Q_A}$,

- if $\underline{p} \leq \overline{P_A}$, by definition we have $\mathbf{C}_{\boldsymbol{\delta}}(\max\{\underline{p}, \underline{P_A}\},\ \underline{Q_A}) \leq \mathbf{C}_{\boldsymbol{\delta}}(x,\ \underline{Q_A})$ for $x \in [\underline{P_A},\ \overline{P_A}]$. According to Proposition 2 and condition $\underline{q} \leq \underline{Q_A}$, we further have $\mathbf{C}_{\boldsymbol{\delta}}(\max\{\underline{p}, \underline{P_A}\},\ \underline{Q_A}) \leq \mathbf{C}_{\boldsymbol{\delta}}(\max\{\underline{p}, \underline{P_A}\}, y) \leq \mathbf{C}_{\boldsymbol{\delta}}(x,\ y)$ for arbitrary $x \in [\underline{P_A}, \overline{P_A}]$ and $y \in [\underline{Q_A}, \overline{Q_A}]$. Given that $(\max\{\underline{p}, \underline{P_A}\},\ \underline{Q_A}) \in [\underline{P_A},\ \overline{P_A}] \times [\underline{Q_A},\ \overline{Q_A}]$, $(\underline{p},\ \underline{Q_A})$ solves Eqn. (11);
- if $\underline{p} > \overline{P_A}$, according to Proposition 1, $\mathbf{C}_{\boldsymbol{\delta}}(\overline{P_A},\ \underline{Q_A}) \leq \mathbf{C}_{\boldsymbol{\delta}}(x,\ \underline{Q_A})$ for $x \in [\underline{P_A},\ \overline{P_A}]$.

  We further show that $\mathbf{C}_{\boldsymbol{\delta}}(\overline{P_A},\ \underline{Q_A}) \leq \mathbf{C}_{\boldsymbol{\delta}}(\overline{P_A},\ y)$ for $y \in [\underline{Q_A},\ \overline{Q_A}]$: assume that this is not true, by Proposition 1, the function $x \mapsto \arg\min_y \mathbf{C}_{\boldsymbol{\delta}}(x,\ y)$ has function value larger than $\underline{Q_A}$ at $x = \overline{P_A}$. Since $\mathbf{C}_{\boldsymbol{\delta}}(0,\ 0)$ is the global minimum, by Proposition 2 on $x \mapsto \arg\min_y \mathbf{C}_{\boldsymbol{\delta}}(x,\ y)$, there exists $x_0 \in [0,\ \overline{P_A}]$ such that $\underline{Q_A} = \arg\min_y \mathbf{C}_{\boldsymbol{\delta}}(x_0,\ y)$. Then, we get

$$\begin{aligned}&\mathbf{C}_{\boldsymbol{\delta}}(x_0,\ \underline{Q_A})\\ \leq&\mathbf{C}_{\boldsymbol{\delta}}(\overline{P_A},\ \arg\min_y \mathbf{C}_{\boldsymbol{\delta}}(\overline{P_A},\ y))\\ \leq&\mathbf{C}_{\boldsymbol{\delta}}(\overline{P_A},\ \underline{Q_A})\end{aligned} \tag{81}$$

  following the similar deduction as in Eqn. (80). Since $x_0 \in [0,\ \overline{P_A}]$, Eqn. (81) implies that $\underline{p}$ should be in $[0,\ \overline{P_A}]$ as well, which violates the branching condition. Thus, $\mathbf{C}_{\boldsymbol{\delta}}(\overline{P_A},\ \underline{Q_A}) \leq \mathbf{C}_{\boldsymbol{\delta}}(\overline{P_A},\ y)$ for $y \in [\underline{Q_A},\ \overline{Q_A}]$.

  Using Proposition 2 for function $y \mapsto \arg\min_x \mathbf{C}_{\boldsymbol{\delta}}(x,\ y)$ in interval $[\underline{Q_A},\ \overline{Q_A}]$ together with Proposition 1, we get $\mathbf{C}_{\boldsymbol{\delta}}(\overline{P_A},\ y) \leq \mathbf{C}_{\boldsymbol{\delta}}(x,\ y)$ for $y \in [\underline{Q_A},\ \overline{Q_A}]$ and $x \in [\underline{P_A},\ \overline{P_A}]$. Thus, $(\overline{P_A},\ \underline{Q_A})$ solves Eqn. (11).

□

G.3. Proof of Theorem 5

Proof of Theorem 5. We first define $r_p(\|\boldsymbol{\epsilon}\|_2) = p(\boldsymbol{\epsilon})$ and $r_q(\|\boldsymbol{\epsilon}\|_2) = q(\boldsymbol{\epsilon})$, then easily seen the concrete expressions of r_p and r_q are:

$$r_p(t) = \frac{1}{(2\sigma'^2)^{d/2-k}\pi^{d/2}} \cdot \frac{\Gamma(d/2)}{\Gamma(d/2-k)}, \tag{82}$$

$$r_q(t) = \frac{\nu}{(2\sigma'^2)^{d/2-k}\pi^{d/2}} \cdot \frac{\Gamma(d/2)}{\Gamma(d/2-k)}, \tag{83}$$

where

$$\nu := \frac{\Gamma(d/2-k)}{\gamma(d/2-k, \frac{T^2}{2\sigma'^2})} > 1 \tag{84}$$

and γ is the lower incomplete Gamma function.

Now we use level-set integration similar as the proof in Lemma F.3 to get the expressions of P, Q, and R respectively. Since $\mathcal{P}$ and $\mathcal{Q}$ are ℓ_2-symmetric, without loss of generality, we let $\boldsymbol{\delta} = (r, 0, \ldots, 0)^\intercal$.

(P).

Suppose $P_T = r_p(T)$.

$$\begin{aligned}&P(\lambda_1, \lambda_2)\\ =&\Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}=\mathcal{N}_{\mathrm{E}}(k,\sigma)}[p(\boldsymbol{\epsilon}-\boldsymbol{\delta}) < \lambda_1 p(\boldsymbol{\epsilon}) + \lambda_2 q(\boldsymbol{\epsilon})]\\ =&\int_{\mathbb{R}^d} \mathbb{I}[p(\boldsymbol{x}-\boldsymbol{\delta}) < \lambda_1 p(\boldsymbol{x}) + \lambda_2 q(\boldsymbol{x})] p(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}\\ =&\int_0^{P_T} y\mathrm{d}y \int_{\substack{p(\boldsymbol{x})=y\\ p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda_1 p(\boldsymbol{x})}} \frac{\mathrm{d}\boldsymbol{x}}{\|\nabla p(\boldsymbol{x})\|_2} +\\ &\int_{P_T}^{\infty} y\mathrm{d}y \int_{\substack{p(\boldsymbol{x})=y\\ p(\boldsymbol{x}-\boldsymbol{\delta})<(\lambda_1+\lambda_2\nu)p(\boldsymbol{x})}} \frac{\mathrm{d}\boldsymbol{x}}{\|\nabla p(\boldsymbol{x})\|_2}\\ =&\int_0^{P_T} y\mathrm{d}y \frac{2\pi^{d/2}}{\Gamma(d/2)} r_p^{-1}(y)^{d-1} \left(-\frac{1}{r_p'(r_p^{-1}(y))}\right) \cdot\\ &\Pr[p(\boldsymbol{x}-\boldsymbol{\delta}) \leq \lambda_1 p(\boldsymbol{x}) \,|\, p(\boldsymbol{x}) = y] +\\ &\int_{P_T}^{\infty} y\mathrm{d}y \frac{2\pi^{d/2}}{\Gamma(d/2)} r_p^{-1}(y)^{d-1} \left(-\frac{1}{r_p'(r_p^{-1}(y))}\right) \cdot\\ &\Pr[p(\boldsymbol{x}-\boldsymbol{\delta}) < (\lambda_1+\lambda_2\nu) p(\boldsymbol{x}) \,|\, p(\boldsymbol{x}) = y]\\ \overset{y=r_p(t)}{=}&\int_T^{\infty} r_p(t)\mathrm{d}t \frac{2\pi^{d/2}}{\Gamma(d/2)} t^{d-1} \cdot\\ &\Pr[p(\boldsymbol{x}-\boldsymbol{\delta}) < \lambda_1 p(\boldsymbol{x}) \,|\, \|\boldsymbol{x}\|_2 = t] +\\ &\int_0^{T} r_p(t)\mathrm{d}t \frac{2\pi^{d/2}}{\Gamma(d/2)} t^{d-1} \cdot\\ &\Pr[p(\boldsymbol{x}-\boldsymbol{\delta}) < (\lambda_1+\lambda_2\nu) p(\boldsymbol{x}) \,|\, \|\boldsymbol{x}\|_2 = t]\\ =&\frac{1}{\Gamma(d/2-k)} \int_{T^2/(2\sigma'^2)}^{\infty} t^{d/2-k-1} \exp(-t)\mathrm{d}t \cdot\\ &\Pr[p(\boldsymbol{x}-\boldsymbol{\delta}) < \lambda_1 p(\boldsymbol{x}) \,|\, \|\boldsymbol{x}\|_2 = \sigma'\sqrt{2t}] +\\ &\frac{1}{\Gamma(d/2-k)} \int_0^{T^2/(2\sigma'^2)} t^{d/2-k-1} \exp(-t)\mathrm{d}t \cdot\\ &\Pr[p(\boldsymbol{x}-\boldsymbol{\delta}) < (\lambda_1+\lambda_2\nu) p(\boldsymbol{x}) \,|\, \|\boldsymbol{x}\|_2 = \sigma'\sqrt{2t}]\end{aligned}$$

$$=\mathbb{E}_{t\sim\Gamma(d/2-k,1)}\begin{cases} u_3(t,\lambda_1), & t\geq T^2/(2\sigma'^2) \\ u_3(t,\lambda_1+\lambda_2\nu). & t<T^2/(2\sigma'^2)\end{cases}$$

Here, $u_3(t,\lambda)=\Pr[p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda p(\boldsymbol{x})\,|\,\|\boldsymbol{x}\|_2=\sigma'\sqrt{2t}]$.

(Q).

Similarly,

$$\begin{aligned}
&Q(\lambda_1,\lambda_2)\\
=&\Pr_{\boldsymbol{\epsilon}\sim\mathcal{Q}=\mathcal{N}^{\mathrm{g}}_{\mathsf{trunc}}(k,T,\sigma)}[p(\boldsymbol{\epsilon}-\boldsymbol{\delta})<\lambda_1 p(\boldsymbol{\epsilon})+\lambda_2 q(\boldsymbol{\epsilon})]\\
=&\int_{\|\boldsymbol{x}\|_2\leq T}\mathbb{I}[p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda_1 p(\boldsymbol{x})+\lambda_2 q(\boldsymbol{x})]q(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\\
=&\int_{\nu P_T}^{\infty} y\mathrm{d}y\int_{\substack{q(\boldsymbol{x})=y\\ p(\boldsymbol{x}-\boldsymbol{\delta})<(\lambda_1+\lambda_2\nu)p(\boldsymbol{x})}}\frac{\mathrm{d}\boldsymbol{x}}{\|\nabla q(\boldsymbol{x})\|_2}\\
\overset{y=r_q(t)}{=}&\int_0^T r_q(t)\mathrm{d}t\frac{2\pi^{d/2}}{\Gamma(d/2)}t^{d-1}\cdot\\
&\qquad\Pr[p(\boldsymbol{x}-\boldsymbol{\delta})<(\lambda_1+\lambda_2\nu)p(\boldsymbol{x})\,|\,\|\boldsymbol{x}\|_2=t]\\
=&\nu\mathbb{E}_{t\sim\Gamma(d/2-k,1)}u_3(t,\lambda_1+\lambda_2\nu)\cdot\mathbb{I}\left[t\leq\frac{T^2}{2\sigma'^2}\right].
\end{aligned}$$

(R).

Now, for R:

$$\begin{aligned}
&R(\lambda_1,\lambda_2)\\
=&\Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}=\mathcal{N}^{\mathrm{g}}(k,\sigma)}[p(\boldsymbol{\epsilon})<\lambda_1 p(\boldsymbol{\epsilon}+\boldsymbol{\delta})+\lambda_2 q(\boldsymbol{\epsilon}+\boldsymbol{\delta})]\\
=&\int_{\mathbb{R}^d}\mathbb{I}[p(\boldsymbol{x})<\lambda_1 p(\boldsymbol{x}+\boldsymbol{\delta})+\lambda_2 q(\boldsymbol{x}+\boldsymbol{\delta})]p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\\
=&\mathbb{E}_{t\sim\Gamma(d/2-k,1)}u_4(t,\lambda_1,\lambda_2).
\end{aligned}$$

Here, $u_4(t,\lambda_1,\lambda_2)=\Pr[\lambda_1 p(\boldsymbol{x}+\boldsymbol{\delta})+\lambda_2 q(\boldsymbol{x}+\boldsymbol{\delta})>r_p(\sigma'\sqrt{2t})\,|\,\|\boldsymbol{x}\|_2=\sigma'\sqrt{2t}]$.

Plugging Lemma G.1 into $P(\lambda_1,\lambda_2)$ and $Q(\lambda_1,\lambda_2)$, and then plugging Lemma G.2 into $R(\lambda_1,\lambda_2)$, we yield the desired expressions in theorem statement. □

Lemma G.1. *Under the condition of Theorem 5, let* $\boldsymbol{\delta}=(r,0,\ldots,0)^\intercal$,

$$\begin{aligned}
&u_3(t,\lambda):=\Pr[p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda p(\boldsymbol{x})\,|\,\|\boldsymbol{x}\|_2=\sigma'\sqrt{2t}]\\
&=\mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{(r+\sigma'\sqrt{2t})^2}{4r\sigma'\sqrt{2t}}-\frac{2k\sigma'^2W(\frac{t}{k}e^{\frac{t}{k}}\lambda^{-\frac{1}{k}})}{4r\sigma'\sqrt{2t}}\right),
\end{aligned}\tag{85}$$

where W is the principal branch of Lambert W function.

Proof of Lemma G.1. $p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda p(\boldsymbol{x})$ means that $r_p(\|\boldsymbol{x}-\boldsymbol{\delta}\|_2)<\lambda r_p(\|\boldsymbol{x}\|_2)$ and therefore $\|\boldsymbol{x}-\boldsymbol{\delta}\|_2>r_p^{-1}(\lambda r_p(\|\boldsymbol{x}\|_2))$. Given that $\|\boldsymbol{x}\|_2=\sigma'\sqrt{2t}$, we have

$$\begin{cases} x_1^2+\sum_{i=2}^d x_i^2=2t\sigma'^2, \\ (x_1-r)^2+\sum_{i=2}^d x_i^2\geq r_p^{-1}(\lambda r_p(\sigma'\sqrt{2t}))^2.\end{cases}\tag{86}$$

This is equivalent to

$$x_1\leq\frac{2t\sigma'^2+r^2-r_p^{-1}(\lambda r_p(\sigma'\sqrt{2t}))^2}{2r}.\tag{87}$$

From the expression of r_p (Eqn. (82)), we have

$$r_p^{-1}(\lambda r_p(\sigma'\sqrt{2t}))^2=2\sigma'^2kW\left(\frac{t}{k}e^{\frac{t}{k}}\lambda^{-\frac{1}{k}}\right).\tag{88}$$

Thus, when $\|\boldsymbol{x}\|_2=\sigma'\sqrt{2t}$ and $\boldsymbol{x}$ uniformly sampled from this sphere,

$$\begin{aligned}
&p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda p(\boldsymbol{x})\\
\iff& x_1\leq\frac{2t\sigma'^2+r^2-2\sigma'^2kW(\frac{t}{k}e^{\frac{t}{k}}\lambda^{-\frac{1}{k}})}{2r}\\
\iff&\frac{1+\frac{x_1}{\sigma'\sqrt{2t}}}{2}\leq\frac{(r+\sigma'\sqrt{2t})^2-2\sigma'^2kW\left(\frac{t}{k}e^{\frac{t}{k}}\lambda^{-\frac{1}{k}}\right)}{4r\sigma'\sqrt{2t}}.
\end{aligned}\tag{89}$$

According to (Yang et al., 2020, Lemma I.23), for $\boldsymbol{x}$ uniformly sampled from sphere with radius $\sigma'\sqrt{2t}$, the component coordinate $\frac{1+\frac{x_1}{\sigma'\sqrt{2t}}}{2}\sim\mathrm{Beta}(\frac{d-1}{2},\frac{d-1}{2})$. Combining Eqn. (89) with this result concludes the proof. □

Lemma G.2. *Under the condition of Theorem 5, let* $\boldsymbol{\delta}=(r,0,\ldots,0)^\intercal$,

$$\begin{aligned}
u_4(t,\lambda_1,\lambda_2):=&\Pr[\lambda_1 p(\boldsymbol{x}+\boldsymbol{\delta})+\lambda_2 q(\boldsymbol{x}+\boldsymbol{\delta})>r_p(\sigma'\sqrt{2t})\\
&\qquad\,|\,\|\boldsymbol{x}\|_2=\sigma'\sqrt{2t}]\\
=&\begin{cases} u_1(t), & \lambda_1\leq 0\\ u_1(t)+u_2(t), & \lambda_1>0\end{cases}
\end{aligned}\tag{90}$$

where

$$\begin{aligned}
u_1(t)=&\mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{\min\{T^2,2\sigma'^2kW(\frac{t}{k}e^{\frac{t}{k}}(\lambda_1+\nu\lambda_2)^{\frac{1}{k}})\}}{4r\sigma'\sqrt{2t}}\right.\\
&\left.-\frac{(\sigma'\sqrt{2t}-r)^2}{4r\sigma'\sqrt{2t}}\right),\\
u_2(t)=&\max\left\{0,\mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{2\sigma'^2kW(\frac{t}{k}e^{\frac{t}{k}}\lambda_1^{\frac{1}{k}})-(\sigma'\sqrt{2t}-r)^2}{4r\sigma'\sqrt{2t}}\right)\right.\\
&\left.-\mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{T^2-(\sigma'\sqrt{2t}-r)^2}{4r\sigma'\sqrt{2t}}\right)\right\},
\end{aligned}$$

Proof of Lemma G.2. Under the condition that $\|\boldsymbol{x}\|_2=\sigma'\sqrt{2t}$, we separate two cases: $q(\boldsymbol{x}+\boldsymbol{\delta})>0$ and $q(\boldsymbol{x}+\boldsymbol{\delta})=0$, which corresponds to $\|\boldsymbol{x}+\boldsymbol{\delta}\|_2\leq T$ and $\|\boldsymbol{x}+\boldsymbol{\delta}\|_2>T$.

(1) $q(\boldsymbol{x}+\boldsymbol{\delta})>0$:

Notice that

$$\begin{aligned} & q(\boldsymbol{x}+\boldsymbol{\delta})>0 \\ \Longleftrightarrow\ & \|\boldsymbol{x}+\boldsymbol{\delta}\|_2^2 \leq T^2 \\ \Longleftrightarrow\ & x_1 \leq \frac{T^2-2t\sigma'^2-r^2}{2r}. \end{aligned} \tag{91}$$

From Eqn. (83), $q(\boldsymbol{x}+\boldsymbol{\delta})=\nu p(\boldsymbol{x}+\boldsymbol{\delta})$. Thus,

$$\begin{aligned} & \lambda_1 p(\boldsymbol{x}+\boldsymbol{\delta})+\lambda_2 q(\boldsymbol{x}+\boldsymbol{\delta})>r_p(\sigma'\sqrt{2t}) \\ \Longleftrightarrow\ & (\lambda_1+\nu\lambda_2)p(\boldsymbol{x}+\boldsymbol{\delta})\geq r_p(\sigma'\sqrt{2t}) \\ \Longleftrightarrow\ & \|\boldsymbol{x}+\boldsymbol{\delta}\|_2^2 \leq r_p^{-1}\left(\frac{r_p(\sigma'\sqrt{2t})}{\lambda_1+\nu\lambda_2}\right)^2 \\ \Longleftrightarrow\ & x_1 \leq \frac{r_p^{-1}\left(\frac{r_p(\sigma'\sqrt{2t})}{\lambda_1+\nu\lambda_2}\right)^2-2t\sigma'^2-r^2}{2r}. \end{aligned} \tag{92}$$

Therefore,

$$\begin{aligned} &\Pr[\lambda_1 p(\boldsymbol{x}+\boldsymbol{\delta})+\lambda_2 q(\boldsymbol{x}+\boldsymbol{\delta})>r_p(\sigma'\sqrt{2t})\wedge q(\boldsymbol{x}+\boldsymbol{\delta})>0 \,|\, \|\boldsymbol{x}\|_2=\sigma'\sqrt{2t}] \\ =\ &\Pr\left[x_1 \leq \frac{\min\left\{r_p^{-1}\left(\frac{r_p(\sigma'\sqrt{2t})}{\lambda_1+\nu\lambda_2}\right)^2, T^2\right\}-2t\sigma'^2-r^2}{2r} \,\middle|\, \|\boldsymbol{x}\|_2=\sigma'\sqrt{2t}\right]. \end{aligned} \tag{93}$$

By (Yang et al., 2020, Lemma I.23) and Eqn. (88), we thus have

$$\begin{aligned} &\Pr[\lambda_1 p(\boldsymbol{x}+\boldsymbol{\delta})+\lambda_2 q(\boldsymbol{x}+\boldsymbol{\delta})>r_p(\sigma'\sqrt{2t}) \\ &\quad\wedge q(\boldsymbol{x}+\boldsymbol{\delta})>0 \,|\, \|\boldsymbol{x}\|_2=\sigma'\sqrt{2t}] \\ =&\mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{\min\{T^2, 2\sigma'^2 kW(\frac{t}{k}e^{\frac{t}{k}}(\lambda_1+\nu\lambda_2)^{\frac{1}{k}})\}}{4r\sigma'\sqrt{2t}}\right. \\ &\left.-\frac{(\sigma'\sqrt{2t}-r)^2}{4r\sigma'\sqrt{2t}}\right)=u_1(t). \end{aligned} \tag{94}$$

(2) $q(\boldsymbol{x}+\boldsymbol{\delta})=0$:

Similarly,

$$q(\boldsymbol{x}+\boldsymbol{\delta})=0 \iff x_1>\frac{T^2-2t\sigma'^2-r^2}{2r}. \tag{95}$$

When $\lambda_1\leq 0$, the condition $\lambda_1 p(\boldsymbol{x}+\boldsymbol{\delta})+\lambda_2 q(\boldsymbol{x}+\boldsymbol{\delta})=\lambda_1 p(\boldsymbol{x}+\boldsymbol{\delta})>r_p(\sigma'\sqrt{2t})$ can never be satisfied. When $\lambda_1>0$, we have

$$\begin{aligned} & \lambda_1 p(\boldsymbol{x}+\boldsymbol{\delta})+\lambda_2 q(\boldsymbol{x}+\boldsymbol{\delta})>r_p(\sigma'\sqrt{2t}) \\ \Longleftrightarrow\ & \|\boldsymbol{x}+\boldsymbol{\delta}\|_2^2 \leq r_p^{-1}\left(\frac{r_p(\sigma'\sqrt{2t})}{\lambda_1}\right)^2 \\ \Longleftrightarrow\ & x_1 \leq \frac{r_p^{-1}\left(\frac{r_p(\sigma'\sqrt{2t})}{\lambda_1}\right)^2-2t\sigma'^2-r^2}{2r}. \end{aligned} \tag{96}$$

Therefore, when $\lambda_1\leq 0$,

$$\begin{aligned} &\Pr[\lambda_1 p(\boldsymbol{x}+\boldsymbol{\delta})+\lambda_2 q(\boldsymbol{x}+\boldsymbol{\delta})>r_p(\sigma'\sqrt{2t}) \\ &\quad\wedge q(\boldsymbol{x}+\boldsymbol{\delta})=0 \,|\, \|\boldsymbol{x}\|_2=\sigma'\sqrt{2t}]=0. \end{aligned} \tag{97}$$

When $\lambda_1>0$, the condition that $lambda_1 p(\boldsymbol{x}+\boldsymbol{\delta})+\lambda_2 q(\boldsymbol{x}+\boldsymbol{\delta})>r_p(\sigma'\sqrt{2t})$ is equivalent to

$$x_1 \in \left(\frac{T^2-2t\sigma'^2-r^2}{2r}, \frac{r_p^{-1}\left(\frac{r_p(\sigma'\sqrt{2t})}{\lambda_1}\right)^2-2t\sigma'^2-r^2}{2r}\right]. \tag{98}$$

Again, by (Yang et al., 2020, Lemma I.23) and Eqn. (88), we have

$$\begin{aligned} &\Pr[\lambda_1 p(\boldsymbol{x}+\boldsymbol{\delta})+\lambda_2 q(\boldsymbol{x}+\boldsymbol{\delta})>r_p(\sigma'\sqrt{2t}) \\ &\quad\wedge q(\boldsymbol{x}+\boldsymbol{\delta})=0 \,|\, \|\boldsymbol{x}\|_2=\sigma'\sqrt{2t}] \\ =&\max\left\{0, \mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{2\sigma'^2 kW(\frac{t}{k}e^{\frac{t}{k}}\lambda_1^{\frac{1}{k}})-(\sigma'\sqrt{2t}-r)^2}{4r\sigma'\sqrt{2t}}\right)\right. \\ &\left.-\mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{T^2-(\sigma'\sqrt{2t}-r)^2}{4r\sigma'\sqrt{2t}}\right)\right\} \\ =&u_2(t). \end{aligned} \tag{99}$$

(3) Combining the two cases:

Now we are ready to combine the two cases.

$$\begin{aligned} &\Pr[\lambda_1 p(\boldsymbol{x}+\boldsymbol{\delta})+\lambda_2 q(\boldsymbol{x}+\boldsymbol{\delta})>r_p(\sigma'\sqrt{2t}) \\ &\quad\,|\, \|\boldsymbol{x}\|_2=\sigma'\sqrt{2t}] \\ =&\Pr[\lambda_1 p(\boldsymbol{x}+\boldsymbol{\delta})+\lambda_2 q(\boldsymbol{x}+\boldsymbol{\delta})>r_p(\sigma'\sqrt{2t}) \\ &\quad\wedge q(\boldsymbol{x}+\boldsymbol{\delta})>0 \,|\, \|\boldsymbol{x}\|_2=\sigma'\sqrt{2t}]+ \\ &\Pr[\lambda_1 p(\boldsymbol{x}+\boldsymbol{\delta})+\lambda_2 q(\boldsymbol{x}+\boldsymbol{\delta})>r_p(\sigma'\sqrt{2t}) \\ &\quad\wedge q(\boldsymbol{x}+\boldsymbol{\delta})=0 \,|\, \|\boldsymbol{x}\|_2=\sigma'\sqrt{2t}] \\ =&\begin{cases} u_1(t), & \lambda_1\leq 0 \\ u_1(t)+u_2(t). & \lambda_1>0 \end{cases} \end{aligned} \tag{100}$$

□

G.4. Discussion on Uniqueness of Feasible Pair

As we sketched in Section 5.3, in general cases, the pair that satisfies constraints in Eqn. (12) is unique. We formally state this finding and prove it in Theorem 7.

Theorem 7 (Uniqueness of Feasible Solution in Eqn. (12))**.** *Under the same setting of Theorem 5, if $Q_A\in(0,1)$ and $P_A\in(Q_A/\nu, 1-(1-Q_A)/\nu)$, then there is the pair (λ_1,λ_2) that satisfies both $P(\lambda_1,\lambda_2)=P_A$ and $Q(\lambda_1,\lambda_2)=Q_A$ is unique.*

We prove the theorem in the end of this section, which is based on the strict monotonicity of two auxiliary functions: $g(\lambda_1+\nu\lambda_2):=Q(\lambda_1,\lambda_2)$ and $h(\lambda_1)$ (defined in

Section 5.3). For other types of smoothing distributions $\mathcal{P}$ and $\mathcal{Q}$, in Theorem 11 we characterize and prove a sufficient condition that guarantees the uniqueness of feasible pair.

We observe that the feasible region of (P_A, Q_A) is

$$\mathcal{R} = \{(x, y) : y/\nu \le x \le 1 - (1-y)/\nu, 0 \le y \le 1\}. \tag{101}$$

Therefore, the theorem states that when (P_A, Q_A) is an internal point of $\mathcal{R}$, the feasible solution is unique and we can use our proposed method to find out such a feasible solution and thus solve the dual problem (**D**). Now, the edge cases are that (P_A, Q_A) lies on the boundary of $\mathcal{R}$. We discuss all these cases and show that these boundary cases correspond to degenerate problems that are easy to solve respectively:

- $Q_A = 0$:
  When $Q_A = 0$, and $P_A \in (0, 1)$ (otherwise, trivially $R(\lambda_1, \lambda_2) = P_A \in \{0, 1\}$ solves (**D**)), we have $\lambda_1 + \nu\lambda_2 \to 0^+$ and thus $R(\lambda_1, \lambda_2) = \mathbb{E}_{t\sim\Gamma(d/2-k,1)} u_2(t)$. Since $u_2(t)$ only involves λ_1, we only require λ_1 to be unique to deploy the method. Since $P_A = P(\lambda_1, \lambda_2) = \mathbb{E}_{t\sim\Gamma(d/2-k,1)} u_3(t, \lambda_1) \cdot \mathbb{I}[t \ge T^2/(2\sigma'^2)]$ and $P_A \in (0, 1)$, by similar arguments as in Theorem 7, we know λ_1 is unique. Hence, all feasible pairs give the same $R(\lambda_1, \lambda_2)$, i.e., have the same objective value and the proposed method that computes a feasible one is sufficient for solving (**D**).
- $Q_A = 1$:
  When $Q_A = 1$ and $P_A \in (0, 1)$, we observe that $u_1(t) \le \mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{T^2}{4r\sigma'\sqrt{2t}} - \frac{(\sigma'\sqrt{2t}-r)^2}{4r\sigma'\sqrt{2t}}\right)$ where equality is feasible with the selected $(\lambda_1 + \nu\lambda_2) \to +\infty$ and hence the maximum of $\mathbb{E}_{t\sim\Gamma(d/2-k,1)} u_1(t)$ among all feasible (λ_1, λ_2) is a constant. On the other hand, since $u_2(t)$ only involves λ_1 that is unique as discussed in "$Q_A = 0$" case, all feasible pairs give the same value of $\mathbb{E}_{t\sim\Gamma(d/2-k,1)} u_2(t)$. As a result, the maximum of $R(\lambda_1, \lambda_2)$ can be computed by adding the unique value of $\mathbb{E}_{t\sim\Gamma(d/2-k,1)} u_2(t)$ and the constant corresponding to the maximum of $\mathbb{E}_{t\sim\Gamma(d/2-k,1)} u_1(t)$ among all feasible (λ_1, λ_2), which solves the dual problem (**D**).
- $P_A = Q_A/\nu$:
  We assume $P_A, Q_A \in (0, 1)$ (otherwise covered by former cases). In this case, λ_1 satisfies that $h(\lambda_1) = P_A - Q_A/\nu = 0$, so $\lambda_1 \to 0^+$. As a result, $u_2(t) = 0$ for all $t > 0$ and $R(\lambda_1, \lambda_2) = \mathbb{E}_{t\sim\Gamma(d/2-k,1)} u_1(t)$. We observe that $u_1(t)$ is only related to $(\lambda_1 + \nu\lambda_2)$ where $(\lambda_1 + \nu\lambda_2)$ satisfying $Q(\lambda_1, \lambda_2) = Q_A$ is unique since $Q_A \in (0, 1)$. Thus, any feasible (λ_1, λ_2) would have the same $(\lambda_1 + \nu\lambda_2)$ and hence leads to the same $R(\lambda_1, \lambda_2)$. So the proposed method that finds one feasible (λ_1, λ_2) suffices for solving (**D**).
- $P_A = 1 - \frac{1-Q_A}{\nu}$:
  We again assume $P_A, Q_A \in (0, 1)$ (otherwise covered by former cases). In this case, λ_1 satisfies that $h(\lambda_1) = 1 - 1/\nu$. Since $\mathbb{E}_{t\sim\Gamma(d/2-k,1)}\mathbb{I}[t < T^2/(2\sigma'^2)] = 1/\nu$, we know $u_3(t, \lambda_1) = 1$ for $t \ge T^2/(2\sigma'^2)$ except a zero-measure set, and thus $\lambda_1 \to +\infty$. As a result, $\mathbb{E}_{t\sim\Gamma(d/2-k,1)} u_2(t) = 1 - \mathbb{E}_{t\sim\Gamma(d/2-k,1)} \mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{T^2-(\sigma'\sqrt{2t}-r)^2}{4r\sigma'\sqrt{2t}}\right)$ is a constant. Similar as "$P_A = Q_A/\nu$" case, feasible $(\lambda_1 + \nu\lambda_2)$ is unique. Therefore, feasible (λ_1, λ_2) leads to a unique $\mathbb{E}_{t\sim\Gamma(d/2-k,1)} u_1(t)$. We compute out these two quantities $\mathbb{E}_{t\sim\Gamma(d/2-k,1)} u_2(t)$ and $\mathbb{E}_{t\sim\Gamma(d/2-k,1)} u_1(t)$ so as to obtain the unique $R(\lambda_1, \lambda_2)$ that solves (**D**).

We remark that in practice, we never observe any instances that correspond to these edge cases though we implemented these techniques for solving them.

Proof of Theorem 7. The high-level proof sketch is implied in the derivation of our feasible (λ_1, λ_2) finding method introduced in Section 5.3. We first show $h(\lambda_1)$ is monotonically strictly increasing in a neighborhood of λ_1 where $h(\lambda_1) = P_A - Q_A/\nu$, so the λ_1 that satisfies $P_A - Q_A/\nu$ is unique. We then define $g(\gamma) = \nu\mathbb{E}_{t\sim\Gamma(d/2-k,1)} u_3(t, \gamma) \cdot \mathbb{I}[t \le T^2/(2\sigma'^2)]$, and show its strict monotonicity around the neighborhood of λ^* that satisfies $g(\gamma^*) = Q_A$, which indicates that $(\lambda_1 + \nu\lambda_2)$ that satisfies Q_A is unique. Combining this two arguments, we know feasible (λ_1, λ_2) is unique. Note that the key in this proof is the local *strict* monotonicity, and the global (nonstrict) monotonicity for both $h(\cdot)$ and $g(\cdot)$ is apparent from their expressions. We now present the proofs for the local strict monotonicity of these two functions h and g.

(1) $h(\lambda_1)$ is monotonically strictly increasing in a neighborhood of λ_1 where $h(\lambda_1) = P_A - Q_A/\nu$.

From the theorem condition, we know that $h(\lambda_1) \in (0, 1 - 1/\nu)$. Thus, there exists $t_0 \ge T^2/(2\sigma'^2)$, such that $u_3(t_0, \lambda_1) \in (0, 1)$ (otherwise $h(\lambda_1) = 0$ or $1 - 1/\nu$). From the closed-form equation of u_3, for any neighboring $\lambda_1' \ne \lambda_1$, we will have $W(t_0/ke^{t_0/k}\lambda_1^{-1/k}) \ne W(t_0/ke^{t_0/k}\lambda_1'^{-1/k})$ by the monotonicity of Lambert W function. On the other hand, since $u_3(t_0, \lambda_1) \in (0, 1)$ and $\mathrm{BetaCDF}_{\frac{d-1}{2}}(\cdot)$ is also strict monotonic in the neighborhood, we have $u_3(t_0, \lambda_1') \ne u_3(t_0, \lambda_1)$. Same happens to t_0's neighborhood, i.e., $\exists\delta > 0$, s.t., $\forall t \in (t_0 - \delta, t_0 + \delta)$, $u_3(t, \lambda_1') \ne u_3(t, \lambda_1)$ and $\mathrm{sgn}(u_3(t, \lambda_1') - u_3(t, \lambda_1))$ is consistent for any $t \in (t_0 - \delta, t_0 + \delta)$. As a result, $h(\lambda_1) \ne h(\lambda_1')$. By definition and the fact that $h(\cdot)$ is monotonically non-decreasing, the argument is proved.

(2) $g(\gamma)$ is monotonically strictly increasing in a neighborhood of γ^* where $g(\gamma^*) = Q_A$.

Since $Q_A \in (0,1)$ by the theorem condition, we know that there exists $t_0 \in (0, T^2/(2\sigma'^2))$, such that $u_3(t_0, \gamma^*) \in (0,1)$ (otherwise $g(\gamma^*) = 0$ or 1, which contradicts the theorem condition). Following the same reasoning as in (1)'s proof, for any $\gamma' \neq \gamma^*$ that lies in a sufficiently small neighborhood of γ^*, we have $u_3(t_0, \gamma^*) \neq u_3(t_0, \gamma')$, and $\exists \delta > 0$, s.t., $\forall t \in (t_0 - \delta, t_0 + \delta)$, $u_3(t, \gamma^*) \neq u_3(t, \gamma')$ and $\operatorname{sgn}(u_3(t, \gamma^*) - u_3(t, \gamma'))$ is consistent for any $t \in (t_0 - \delta, t_0 + \delta)$. As a result, $g(\gamma^*) \neq g(\gamma')$. By definition and the fact that $g(\cdot)$ is monotonically non-decreasing, the argument is proved. □

H. Extensions of DSRS Computational Methods

In this appendix, we exemplify a few extensions of DSRS computational method.

H.1. Certification with Standard and Truncated Standard Gaussian

In main text and Theorem 5, we focus on DSRS certification with generalized Gaussian as $\mathcal{P}$ and truncated generalized Gaussian as $\mathcal{Q}$, which has theoretical advantages (Theorem 2). On the other hand, DSRS can also be applied to other distributions. Concretely, to certify robustness with standard Gaussian as $\mathcal{P}$ and truncated standard Gaussian as $\mathcal{Q}$, we can directly plug the following theorem's numerical integration expressions into the described DSRS algorithm (Alg. 2).

Theorem 8. *In $\mathbf{D}_{\boldsymbol{\delta}}(P_A, Q_A)$, let $r = \|\boldsymbol{\delta}\|_2$, when $\mathcal{P} = \mathcal{N}(\sigma)$ and $\mathcal{Q} = \mathcal{N}_{\mathsf{trunc}}(T, \sigma)$, let $\nu := \Gamma\mathrm{CDF}_{d/2}(T^2/(2\sigma^2))^{-1}$,*

$$
\begin{aligned}
R(\lambda_1, \lambda_2) &:= \Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[p(\boldsymbol{\epsilon}) < \lambda_1 p(\boldsymbol{\epsilon} + \boldsymbol{\delta}) + \lambda_2 q(v\epsilon + \boldsymbol{\delta})] \\
&= \begin{cases} \mathbb{E}_{t\sim\Gamma(d/2,1)} u_1(t), & \lambda_1 \le 0 \\ \mathbb{E}_{t\sim\Gamma(d/2,1)} u_1(t) + u_2(t), & \lambda_1 > 0 \end{cases} \quad \text{where} \\
u_1(t) &= \mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{\min\{T^2, 2t\sigma^2 + 2\sigma^2 \ln(\lambda_1 + \nu\lambda_2)\}}{4r\sigma\sqrt{2t}} - \frac{(\sigma\sqrt{2t} - r)^2}{4r\sigma\sqrt{2t}}\right), \\
u_2(t) &= \max\left\{0, \mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{2t\sigma^2 + 2\sigma^2 \ln\lambda_1 - (\sigma\sqrt{2t} - r)^2}{4r\sigma\sqrt{2t}}\right) - \mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{T^2 - (\sigma\sqrt{2t} - r)^2}{4r\sigma\sqrt{2t}}\right)\right\}.
\end{aligned}
$$

$$
\begin{aligned}
P(\lambda_1, \lambda_2) &:= \Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[p(\boldsymbol{\epsilon} - \boldsymbol{\delta}) < \lambda_1 p(\boldsymbol{\epsilon}) + \lambda_2 q(\boldsymbol{\epsilon})] \\
&= \mathbb{E}_{t\sim\Gamma(d/2,1)} \begin{cases} u_3(t, \lambda_1), & t \ge T^2/(2\sigma^2) \\ u_3(t, \lambda_1 + \nu\lambda_2), & t < T^2/(2\sigma^2). \end{cases} \quad \text{where} \\
u_3(t, \lambda) &= \mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{1}{2} + \frac{r^2 + 2\sigma^2 \ln\lambda}{4r\sigma\sqrt{2t}}\right).
\end{aligned}
$$

$$
\begin{aligned}
Q(\lambda_1, \lambda_2) &:= \Pr_{\boldsymbol{\epsilon}\sim\mathcal{Q}}[p(\boldsymbol{\epsilon} - \boldsymbol{\delta}) < \lambda_1 p(\boldsymbol{\epsilon}) + \lambda_2 q(\boldsymbol{\epsilon})] \\
&= \nu\mathbb{E}_{t\sim\Gamma(d/2,1)} u_3(t, \lambda_1 + \nu\lambda_2) \cdot \mathbb{I}[t \le T^2/(2\sigma^2)].
\end{aligned}
$$

In above equations, $\Gamma(d/2, 1)$ is gamma distribution and $\Gamma\mathrm{CDF}_{d/2}$ is its CDF, and $\mathrm{BetaCDF}_{\frac{d-1}{2}}$ is the CDF of distribution $\mathrm{Beta}(\frac{d-1}{2}, \frac{d-1}{2})$.

Proof of Theorem 8. The proof largely follows the same procedure as the proof of Theorem 5. The only difference is that, since $\mathcal{P} = \mathcal{N}(\sigma)$, let $r_p(\|\boldsymbol{\epsilon}\|_2) = p(\boldsymbol{\epsilon})$, different from Eqn. (88), now

$$
\begin{aligned}
& r_p^{-1}(\lambda r_p(\sigma\sqrt{2t}))^2 \\
=& -2\sigma^2 \ln(\lambda r_p(\sigma\sqrt{2t}) \cdot (2\pi\sigma^2)^{d/2}) \\
=& -2\sigma^2 \ln\left(\frac{\lambda}{(2\pi\sigma^2)^{d/2}} \exp\left(-\frac{2t\sigma^2}{2\sigma^2}\right) \cdot (2\pi\sigma^2)^{d/2}\right) \\
=& -2\sigma^2(\ln\lambda - t) = 2t\sigma^2 - 2\sigma^2 \ln\lambda.
\end{aligned} \tag{103}
$$

By plugging this equation into the proof of Theorem 5, we prove Theorem 8. □

In practice, DSRS with standard Gaussian and truncated standard Gaussian as smoothing distributions gives marginal improvements over Neyman-Pearson-based certification. This is because, for standard Gaussian distribution, the noise magnitude is particularly concentrated on a thin shell as reflected by the green curve in Figure 4. As a result, the truncated standard Gaussian as $\mathcal{Q}$ either has a tiny density overlap with $\mathcal{P}$ or provides highly similar information (i.e., $Q_A \approx P_A$). In either case, $\mathcal{Q}$ provides little additional information. Therefore, in practice, we do not use standard Gaussian and truncated standard Gaussian as $\mathcal{P}$ and $\mathcal{Q}$, which is also justified by Theorem 2, though DSRS can provide certification for this setting.

H.2. Certification with Generalized Gaussian with Different Variances

We now consider the robustness certification with smoothing distribution $\mathcal{P} = \mathcal{N}^{\mathrm{g}}(k, \sigma)$ and additional smoothing distribution $\mathcal{Q} = \mathcal{N}^{\mathrm{g}}(k, \beta)$ where σ and β are different (i.e., different variance).

H.2.1. Computational Method Description

Hereinafter, for this $\mathcal{P}$ and $\mathcal{Q}$ we define the radial density function $g(r) := p(\boldsymbol{x})$ and $h(r) := q(\boldsymbol{x})$ for any $\|\boldsymbol{x}\|_2 = r$, where p and q are the density functions of $\mathcal{P}$ and $\mathcal{Q}$ respectively. $\overline{(\lambda_1 g + \lambda_2 h)^{-1}}(x) := \max y \quad \text{s.t. } \lambda_1 g(y) + \lambda_2 h(y) = x$ and similarly $\underline{(\lambda_1 g + \lambda_2 h)^{-1}}(x) := \min y \quad \text{s.t. } \lambda_1 g(y) + \lambda_2 h(y) = x$.

In this case, we can still have the numerical expression for $P(\lambda_1, \lambda_2)$, $Q(\lambda_1, \lambda_2)$, and $R(\lambda_1, \lambda_2)$ as shown in Theorem 9.

Theorem 9. *When the smoothing distributions $\mathcal{P} = \mathcal{N}^{\mathrm{g}}(k, \sigma)$ and additional smoothing distribution $\mathcal{Q} =$*

Table 3. The numerical integration expression for P, Q, and R (see definition in Theorem 5). See Appendix H.2 for notation description.

$$P(\lambda_1,\,\lambda_2) = \mathop{\mathbb{E}}_{x\sim\Gamma(\frac{d}{2}-k)} \mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{(r+\sigma'\sqrt{2x})^2 - g^{-1}(\lambda_1 g(\sigma'\sqrt{2x}) + \lambda_2 h(\sigma'\sqrt{2x}))^2}{4r\sigma'\sqrt{2x}}\right)$$

$$Q(\lambda_1,\,\lambda_2) = \mathop{\mathbb{E}}_{x\sim\Gamma(\frac{d}{2}-k)} \mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{(r+\beta'\sqrt{2x})^2 - g^{-1}(\lambda_1 g(\beta'\sqrt{2x}) + \lambda_2 h(\beta'\sqrt{2x}))^2}{4r\beta'\sqrt{2x}}\right)$$

$$R(\lambda_1,\,\lambda_2) = \mathop{\mathbb{E}}_{x\sim\Gamma(\frac{d}{2}-k)} \mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{1}{2} + \frac{\overline{(\lambda_1 g + \lambda_2 h)^{-1}}(g(\sigma'\sqrt{2x}))^2 - r^2 - 2x\sigma'^2}{4r\sigma'\sqrt{2x}}\right) - \mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{1}{2} + \frac{\overline{(\lambda_1 g + \lambda_2 h)^{-1}}(g(\sigma'\sqrt{2x}))^2 - r^2 - 2x\sigma'^2}{4r\sigma'\sqrt{2x}}\right)$$

Table 4. Simplified terms in numerical integration for P and Q, where W is the real-valued branch of the Lambert W function.

Functions		$P(\lambda_1,\,\lambda_2)$	$Q(\lambda_1,\,\lambda_2)$
Term to Simplify		$g^{-1}\left(\lambda_1 g(\sigma'\sqrt{2x}) + \lambda_2 h(\sigma'\sqrt{2x})\right)^2$	$g^{-1}\left(\lambda_1 g(\beta'\sqrt{2x}) + \lambda_2 h(\beta'\sqrt{2x})\right)^2$
Simplified Terms	$k=0$	$-2\sigma'^2 \ln\left(\lambda_1 \exp(-x) + \lambda_2 \left(\frac{\sigma'}{\beta'}\right)^d \exp\left(-\frac{\sigma'^2}{\beta'^2}x\right)\right)$	$-2\sigma'^2 \ln\left(\lambda_1 \exp\left(-\frac{\beta'^2}{\sigma'^2}x\right) + \lambda_2 \left(\frac{\sigma'}{\beta'}\right)^d \exp(-x)\right)$
	$k>0$	$2k\sigma'^2 W\left(\frac{x}{k}\left(\lambda_1 \exp(-x) + \lambda_2 \left(\frac{\sigma'}{\beta'}\right)^{d-2k} \exp\left(-\frac{\sigma'^2}{\beta'^2}x\right)\right)^{-1/k}\right)$	$2k\sigma'^2 W\left(\frac{x}{k}\cdot\frac{\beta'^2}{\sigma'^2}\left(\lambda_1 \exp\left(-\frac{\beta'^2}{\sigma'^2}\right) + \lambda_2 \left(\frac{\sigma'}{\beta'}\right)^{d-2k} \exp(-x)\right)^{-1/k}\right)$

$\mathcal{N}^{\mathtt{g}}(k,\beta)$, let $P(\lambda_1,\lambda_2)$, $Q(\lambda_1,\lambda_2)$, and $R(\lambda_1,\lambda_2)$ be as defined in Theorem 5, then P, Q, and R can be computed by expressions in Table 3.

In Table 3, the numerical integration requires the computation of several inverse functions. In this subsection, we simplify the numerical integration expressions for P and Q by deriving the closed forms of these inverse functions, as shown in Table 4. In the actual implementation of numerical integration, for P and Q, we use these simplified expressions to compute; for R, benefited from the unimodality (Lemma H.2), we deploy a simple binary search to compute.

Theorem 10. *When the smoothing distributions $\mathcal{P} = \mathcal{N}^{\mathtt{g}}(k,\sigma)$ and $\mathcal{Q} = \mathcal{N}^{\mathtt{g}}(k,\beta)$, the terms $g^{-1}(\lambda_1 g(\sigma'\sqrt{2x}) + \lambda_2 h(\sigma'\sqrt{2x}))^2$ and $g^{-1}(\lambda_1 g(\beta'\sqrt{2x}) + \lambda_2 h(\beta'\sqrt{2x}))^2$ in $P(\lambda_1,\,\lambda_2)$ and $Q(\lambda_1,\,\lambda_2)$'s computational expressions (see Table 3) are equivalent to those shown in Table 4.*

With the method to compute P, Q, and R for given λ_1 and λ_2, now the challenge is to solve λ_1 and λ_2 such that $P(\lambda_1,\lambda_2) = P_A$ and $Q(\lambda_1,\lambda_2) = Q_A$.

Luckily, as Theorem 11 shows, for given P_A and Q_A, such (λ_1,λ_2) pair is unique. Indeed, such uniqueness holds not only for this $\mathcal{P}$ and $\mathcal{Q}$ but also for a wide range of smoothing distributions.

Theorem 11 (Uniqueness)**.** *Suppose distributions $\mathcal{P}$ and $\mathcal{Q}$'s are ℓ_p-spherically symmetric, i.e., there exists radial density functions g and h such that $p(\boldsymbol{x}) = g(\|\boldsymbol{x}\|_p)$ and $q(\boldsymbol{x}) = h(\|\boldsymbol{x}\|_p)$, then if g and h are continuous and $\frac{g}{h}$ is continuous and strictly monotonic almost everywhere, for any given $(P_A,\,Q_A) \in \mathbb{R}^2_+$, there is at most one $(\lambda_1,\,\lambda_2)$ pair satisfying constraint of Eqn. (12).*

The proof is shown in the next subsection, which is based on Cauchy's mean value theorem of the probability integral.

With Theorem 11 and Proposition 2, we can use joint binary search as shown in Alg. 4 to find λ_1 and λ_2 that can be viewed as the intersection of two curves. At a high level, Each time, we leverage the monotonicity to get a point $(\lambda_1^{mid},\,\lambda_2^{mid})$ on the P's curve, then compute corresponding Q, and update the binary search intervals based on whether $Q(\lambda_1^{mid},\,\lambda_2^{mid}) > Q_A$. We shrink the intervals for both λ_1 and λ_2 (Lines 5 and 7) in Alg. 4 to accelerate the search. The algorithm is plugged into Line 8 of Alg. 2.

Algorithm 4 DUALBINARYSEARCH for λ_1 and λ_2.

Data: Query access to $P(\cdot,\,\cdot)$ and $Q(\cdot,\,\cdot)$, P_A and Q_A
Result: λ_1 and λ_2 satisfying constraints $P(\lambda_1,\lambda_2) = P_A, Q(\lambda_1,\lambda_2) = Q_A$

1 $\lambda_1^L \leftarrow -M, \lambda_1^U \leftarrow +M, \lambda_2^L \leftarrow -M, \lambda_2^U \leftarrow +M$; /* M is a large positive number */
2 **while** $\lambda_1^U - \lambda_1^L > eps$ **do**
3 $\lambda_1^{mid} \leftarrow (\lambda_1^L + \lambda_1^U)/2$ Binary search for $\lambda_2^{mid} \in [\lambda_2^L,\,\lambda_2^U]$ such that $P(\lambda_1^{mid},\,\lambda_2^{mid}) = P_A$; /* $(\lambda_1^{mid},\,\lambda_2^{mid})$ lies on red curve */
4 **if** $Q(\lambda_1^{mid},\,\lambda_2^{mid}) < Q_A$ **then**
5 $\lambda_1^U \leftarrow \lambda_1^{mid}, \lambda_2^L \leftarrow \lambda_2^{mid}$; /* $(\lambda_1^{mid},\,\lambda_2^{mid})$ is right to intersection */
6 **else**
7 $\lambda_1^L \leftarrow \lambda_1^{mid}, \lambda_2^U \leftarrow \lambda_2^{mid}$; /* $(\lambda_1^{mid},\,\lambda_2^{mid})$ is left to intersection */
8 **end**
9 **return** $(\lambda_1^L,\,\lambda_2^L)$; /* for soundness: $R(\lambda_1^L,\,\lambda_2^L)$ lower bounds (**D**) */

H.2.2. PROOFS

Proof of Theorem 9. The ℓ_2-radial density functions of $p(\boldsymbol{x})$ and $q(\boldsymbol{x})$ have these expressions: $g(r) \propto r^{-k}\exp(-r^2/(2\sigma'^2))$ and $h(r) \propto r^{-k}\exp(-r^2/(2\beta'^2))$. When r increases, r^{-k}, $\exp(-r^2/(2\sigma'^2))$, and $\exp(-r^2/(2\beta'^2))$ decrease so that g and h are both strictly decreasing. Therefore, they have inverse functions, which are denoted by g^{-1} and h^{-1}. Now we are ready to

derive the expressions.

(I.) We start with P:

$$
\begin{aligned}
&P(\lambda_1,\,\lambda_2)\\
=&\int_{\mathbb{R}^d}\mathbb{I}\left[p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda_1 p(\boldsymbol{x})+\lambda_2 q(\boldsymbol{x})\right]p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\\
\overset{(1.)}{=}&\int_0^\infty y\mathrm{d}y\int_{\substack{p(\boldsymbol{x})=y\\ p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda_1 p(\boldsymbol{x})+\lambda_2 h(\boldsymbol{x})}}\frac{\mathrm{d}\boldsymbol{x}}{\|\nabla p(\boldsymbol{x})\|_2}\\
\overset{(2.)}{=}&\int_0^\infty y\mathrm{d}y\int_{\substack{p(\boldsymbol{x})=y\\ p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda_1 p(\boldsymbol{x})+\lambda_2 h(\boldsymbol{x})}}-\frac{\mathrm{d}\boldsymbol{x}}{g'\left(g^{-1}(y)\right)}\\
\overset{(3.)}{=}&\int_0^\infty y\mathrm{d}y\cdot\frac{\pi^{d/2}dg^{-1}(y)^{d-1}}{(d/2)!}\left(-\frac{\mathrm{d}\boldsymbol{x}}{g'(g^{-1}(y))}\right)\cdot\\
&\quad\Pr\left[p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda_1 p(\boldsymbol{x})+\lambda_2 q(\boldsymbol{x})\,\middle|\,p(\boldsymbol{x})=y\right]\\
\overset{(4.)}{=}&\int_0^\infty g(t)\mathrm{d}t\cdot\frac{\pi^{d/2}dt^{d-1}}{(d/2)!}\cdot\\
&\quad\Pr\left[p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda_1 g(t)+\lambda_2 h(t)\,\middle|\,\|\boldsymbol{x}\|_2=t\right]\\
\overset{(5.)}{=}&\int_0^\infty\frac{1}{(2\sigma'^2)^{\frac{d}{2}-k}\pi^{\frac{d}{2}}}\cdot\frac{\Gamma(d/2)}{\Gamma(d/2-k)}\cdot\\
&\quad t^{-2k}\exp\left(-\frac{t^2}{2\sigma'^2}\right)\cdot\frac{\pi^{d/2}dt^{d-1}}{(d/2)!}\cdot\\
&\quad\Pr\left[p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda_1 g(t)+\lambda_2 h(t)\,\middle|\,\|\boldsymbol{x}\|_2=t\right]\\
=&\int_0^\infty\frac{2}{(2\sigma'^2)^{\frac{d}{2}-k}\Gamma(\frac{d}{2}-k)}t^{d-2k-1}\exp\left(-\frac{t^2}{2\sigma'^2}\right)\cdot\\
&\quad\Pr\left[p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda_1 g(t)+\lambda_2 h(t)\,\middle|\,\|\boldsymbol{x}\|_2=t\right]\\
\overset{(6.)}{=}&\frac{1}{(2\sigma'^2)^{\frac{d}{2}-k}\Gamma(\frac{d}{2}-k)}\int_0^\infty t^{\frac{d}{2}-k-1}\exp\left(-\frac{t}{2\sigma'^2}\right)\cdot\\
&\quad\Pr\left[p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda_1 g(\sqrt{t})+\lambda_2 h(\sqrt{t})\,\middle|\,\|\boldsymbol{x}\|_2=\sqrt{t}\right]\\
\overset{(7.)}{=}&\frac{1}{\Gamma(\frac{d}{2}-k)}\int_0^\infty t^{\frac{d}{2}-2k-1}\exp(-t)\cdot\\
&\quad\Pr\Big[p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda_1 g(\sigma'\sqrt{2t})+\lambda_2 h(\sigma'\sqrt{2t})\\
&\qquad\qquad\Big|\,\|\boldsymbol{x}\|_2=\sigma'\sqrt{2t}\Big]\\
\overset{(8.)}{=}&\mathbb{E}_{t\sim\Gamma(\frac{d}{2}-k)}\Pr\Big[p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda_1 g(\sigma'\sqrt{2t})+\lambda_2 h(\sigma'\sqrt{2t})\\
&\qquad\qquad\Big|\,\|\boldsymbol{x}\|_2=\sigma'\sqrt{2t}\Big].
\end{aligned}
$$

As a reminder, d is the input dimension. The $\Gamma(\cdot)$ here refer to either the Gamma distribution (in $t\sim\Gamma(\frac{d}{2}-k)$) or Gamma function (in some denominators). In the above integration: $(1.)$ uses level-set sliced integration as first proposed in (Yang et al., 2020); $(2.)$ leverages the fact that $p(\boldsymbol{x})$ is ℓ_2-symmetric and $g'(\cdot)<0$; $(3.)$ injects the surface area of ℓ_2-sphere with radius $g^{-1}(y)$; $(4.)$ alters the integral variable: $t=g^{-1}(y)$, which yields $\mathrm{d}t=\mathrm{d}y/g'(t)=\mathrm{d}y/g'(g^{-1}(y))$ and $y=g(t)$; $(5.)$ injects the expression of $g(t)$; $(6.)$ alters the integral variable from t to t^2; $(7.)$ does re-scaling; and $(8.)$ observes that the integral can be expressed by expectation over Gamma distribution.

Due to the isotropy, let $r=\|\boldsymbol{\delta}\|_2$, we can deem $\boldsymbol{\delta}=(r,\,0,\,\ldots,\,0)^\intercal$ by rotating the axis. Then we simplify the probability term by observing that

$$
\begin{aligned}
&\begin{cases}\|\boldsymbol{x}\|_2=\sigma'\sqrt{2t}\\ p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda_1 g(\sigma'\sqrt{2t})+\lambda_2 h(\sigma'\sqrt{2t})\end{cases}\\
\iff&\begin{cases}x_1^2+\sum_{i=2}^d x_i^2=2t\sigma'^2\\ (x_1-r)^2+\sum_{i=2}^d x_i^2\geq g^{-1}\left(\lambda_1 g(\sigma'\sqrt{2t})+\lambda_2 h(\sigma'\sqrt{2t})\right)^2\end{cases}\\
\implies&x_1\leq\frac{2t\sigma'^2-g^{-1}\left(\lambda_1 g(\sigma'\sqrt{2t})+\lambda_2 h(\sigma'\sqrt{2t})\right)^2+r^2}{2r}.
\end{aligned}
$$

Lemma H.1 (Lemma I.23; (Yang et al., 2020))**.** *If $(x_1,\cdots,x_d)$ is sampled uniformly from the unit sphere $\mathbb{S}^{d-1}\subseteq\mathbb{R}^d$, then*

$$
\frac{1+x_1}{2}\ \textit{is distributed as}\ \mathrm{Beta}\left(\frac{d-1}{2},\,\frac{d-1}{2}\right).\quad(104)
$$

Combining Lemma H.1 and Appendix H.2.2, we get

$$
\begin{aligned}
&\Pr\left[p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda_1 g(\sigma'\sqrt{2t})+\lambda_2 h(\sigma'\sqrt{2t})\,\middle|\,\|\boldsymbol{x}\|_2=\sigma'\sqrt{2t}\right]\\
=&\mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{(r+\sigma'\sqrt{2t})^2}{4r\sigma'\sqrt{2t}}\right.\\
&\qquad\left.-\frac{g^{-1}\left(\lambda_1 g(\sigma'\sqrt{2t})+\lambda_2 h(\sigma'\sqrt{2t})\right)^2}{4r\sigma'\sqrt{2t}}\right).
\end{aligned}
\quad(105)
$$

Injecting Eqn. (105) into $(8.)$ yields the expression shown in Table 3.

(II.) The integration for Q is similar:

$$
\begin{aligned}
&Q(\lambda_1,\,\lambda_2)\\
=&\int_{\mathbb{R}^d}\mathbb{I}[p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda_1 p(\boldsymbol{x})+\lambda_2 q(x)]q(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\\
=&\int_0^\infty y\mathrm{d}y\int_{\substack{q(\boldsymbol{x})=y\\ p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda_1 p(\boldsymbol{x})+\lambda_2 q(\boldsymbol{x})}}\frac{\mathrm{d}\boldsymbol{x}}{\|\nabla q(\boldsymbol{x})\|_2}\\
=&\int_0^\infty y\mathrm{d}y\int_{\substack{q(\boldsymbol{x})=y\\ p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda_1 p(\boldsymbol{x})+\lambda_2 q(\boldsymbol{x})}}-\frac{\mathrm{d}\boldsymbol{x}}{h'(h^{-1}(y))}\\
=&\int_0^\infty h(t)\mathrm{d}t\cdot\frac{\pi^{d/2}dt^{d-1}}{(d/2)!}\cdot\\
&\quad\Pr\left[p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda_1 p(\boldsymbol{x})+\lambda_2 q(\boldsymbol{x})\,\middle|\,\|\boldsymbol{x}\|_2=t\right]\\
=&\frac{1}{(2\beta'^2)^{\frac{d}{2}-k}\Gamma(\frac{d}{2}-k)}\int_0^\infty t^{\frac{d}{2}-k-1}\exp\left(-\frac{t}{2\beta'^2}\right)\cdot
\end{aligned}
$$

$$\begin{aligned}&\Pr\left[p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda_1 g(\sqrt{t})+\lambda_2 h(\sqrt{t})\,\middle|\,\|\boldsymbol{x}\|_2=\sqrt{t}\right]\\=&\mathbb{E}_{t\sim\Gamma(\frac{d}{2}-k)}\Pr\left[p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda_1 g(\beta'\sqrt{2t})+\lambda_2 h(\beta'\sqrt{2t})\right.\\&\left.\qquad\qquad\,\middle|\,\|\boldsymbol{x}\|_2=\beta'\sqrt{2t}\right],\end{aligned}$$

where

$$\begin{aligned}&\Pr\left[p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda_1 g(\beta'\sqrt{2t})+\lambda_2 h(\beta'\sqrt{2t})\,\middle|\,\|\boldsymbol{x}\|_2=\beta'\sqrt{2t}\right]\\=&\mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{(r+\beta'\sqrt{2t})^2}{4r\beta'\sqrt{2t}}\right.\\&\left.-\frac{g^{-1}\left(\lambda_1 g(\beta'\sqrt{2t})+\lambda_2 h(\beta'\sqrt{2t})\right)^2}{4r\beta'\sqrt{2t}}\right).\end{aligned}\tag{106}$$

(III.) Finally, we derive the integration for R:

$$\begin{aligned}&R(\lambda_1,\,\lambda_2)\\=&\int_{\mathbb{R}^d}\mathbb{I}\left[p(\boldsymbol{x}-\boldsymbol{\delta})<\lambda_1 p(\boldsymbol{x})+\lambda_2 q(\boldsymbol{x})\right]p(\boldsymbol{x}-\boldsymbol{\delta})\mathrm{d}\boldsymbol{x}\\=&\int_{\mathbb{R}^d}\mathbb{I}\left[p(\boldsymbol{x})<\lambda_1 p(\boldsymbol{x}+\boldsymbol{\delta})+\lambda_2 q(\boldsymbol{x}+\boldsymbol{\delta})\right]p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\\=&\int_0^\infty y\mathrm{d}y\int_{\substack{p(\boldsymbol{x})=y\\ p(\boldsymbol{x})<\lambda_1 p(\boldsymbol{x}+\boldsymbol{\delta})+\lambda_2 q(\boldsymbol{x}+\boldsymbol{\delta})}}\frac{\mathrm{d}\boldsymbol{x}}{\|\nabla p(\boldsymbol{x})\|_2}\\=&\int_0^\infty y\mathrm{d}y\int_{\substack{p(\boldsymbol{x})=y\\ p(\boldsymbol{x})<\lambda_1 p(\boldsymbol{x}+\boldsymbol{\delta})+\lambda_2 q(\boldsymbol{x}+\boldsymbol{\delta})}}-\frac{\mathrm{d}\boldsymbol{x}}{g'(g^{-1}(y))}\\=&\int_0^\infty g(t)\mathrm{d}t\cdot\frac{\pi^{d/2}dt^{d-1}}{(d/2)!}\cdot\\&\Pr\left[\lambda_1 p(\boldsymbol{x}+\boldsymbol{\delta})+\lambda_2 q(\boldsymbol{x}+\boldsymbol{\delta})>p(\boldsymbol{x})\,\middle|\,\|\boldsymbol{x}\|_2=t\right]\\=&\frac{1}{(2\sigma'^2)^{\frac{d}{2}-k}\Gamma(\frac{d}{2}-k)}\int_0^\infty t^{\frac{d}{2}-k-1}\exp\left(-\frac{t}{2\sigma'^2}\right)\cdot\\&\Pr\left[\lambda_1 p(\boldsymbol{x}+\boldsymbol{\delta})+\lambda_2 q(\boldsymbol{x}+\boldsymbol{\delta})>g(\sqrt{t})\,\middle|\,\|\boldsymbol{x}\|_2=\sqrt{t}\right]\\\overset{(9.)}{=}&\mathbb{E}_{t\sim\Gamma(\frac{d-k}{2})}\Pr\left[\lambda_1 p(\boldsymbol{x}+\boldsymbol{\delta})+\lambda_2 q(\boldsymbol{x}+\boldsymbol{\delta})>g(\sigma'\sqrt{2t})\right.\\&\left.\qquad\,\middle|\,\|\boldsymbol{x}\|_2=\sigma'\sqrt{2t}\right].\end{aligned}$$

To simplify the probability term, this time we have

$$\begin{cases}\|\boldsymbol{x}\|_2=\sigma'\sqrt{2t},\\ \lambda_1 p(\boldsymbol{x}+\boldsymbol{\delta})+\lambda_2 q(\boldsymbol{x}+\boldsymbol{\delta})>g(\sigma'\sqrt{2t})\end{cases}\tag{107}$$

$$\iff\begin{cases}x_1^2+\sum_{i=2}^d x_i^2=2t\sigma'^2\\ (\lambda_1 g+\lambda_2 h)\left(\sqrt{(x_1+r)^2+\sum_{i=2}^d x_i^2}\right)>g(\sigma'\sqrt{2t})\end{cases}\tag{108}$$

$$\implies(\lambda_1 g+\lambda_2 h)\left(\sqrt{2t\sigma'^2+r^2+2x_1 r}\right)>g(\sigma'\sqrt{2t}),\tag{109}$$

where $r=\|\boldsymbol{\delta}\|_2$ and we deem $\boldsymbol{\delta}=(r,\,0,\,\ldots,\,0)^\intercal$ by rotating the axis.

Lemma H.2. *The function $(\lambda_1 g+\lambda_2 h)$ is unimodal in its domain $(0,\,+\infty)$.*

Proof of Lemma H.2. We expand $(\lambda_1 g+\lambda_2 h)$ then consider its derivative.

$$\begin{aligned}&\lambda_1 g(r)+\lambda_2 h(r)\\=&C_1\lambda_1 r^{-2k}\exp\left(-\frac{r^2}{2\sigma'^2}\right)+C_2\lambda_2 r^{-2k}\exp\left(-\frac{r^2}{2\beta'^2}\right),\end{aligned}\tag{110}$$

where C_1 and C_2 is the constant normalization coefficient of g and h respectively.

$$\begin{aligned}&(\lambda_1 g+\lambda_2 h)'(r)\\=&-\left(C_1\lambda_1 2kr^{-2k-1}\exp\left(-\frac{r^2}{2\sigma'^2}\right)\right.\\&+C_1\lambda_1\cdot\frac{r^{-2k+1}}{\sigma'^2}\exp\left(-\frac{r^2}{2\sigma'^2}\right)\\&+C_2\lambda_2 2kr^{-2k-1}\exp\left(-\frac{r^2}{2\beta'^2}\right)\\&\left.+\,C_2\lambda_2\cdot\frac{r^{-2k+1}}{\beta'^2}\exp\left(-\frac{r^2}{2\beta'^2}\right)\right).\end{aligned}\tag{111}$$

Now we show that

$$(\lambda_1 g+\lambda_2 h)'(r)=0\tag{112}$$

has at most one solution.

When either $\lambda_1=0$ or $\lambda_2=0$, since both g and h are monotonic, $\lambda_1 g+\lambda_2$ is monotonic and there is no solution. Thus, we assume $\lambda_1,\,\lambda_2\neq 0$. We observe that

$$(\lambda_1 g+\lambda_2 h)'(r)=0\tag{113}$$

$$\begin{aligned}\iff&C_1\lambda_1\left(2k+\frac{r^2}{\sigma'^2}\right)\exp\left(-\frac{r^2}{2\sigma'^2}\right)\\&+C_2\lambda_2\left(2k+\frac{r^2}{\beta'^2}\right)\exp\left(-\frac{r^2}{2\beta'^2}\right)=0\end{aligned}\tag{114}$$

$$\iff-\frac{C_1\lambda_1}{C_2\lambda_2}\cdot\frac{2k+\frac{r^2}{\sigma'^2}}{2k+\frac{r^2}{\beta'^2}}=\exp\left(\frac{r^2}{2\sigma'^2}-\frac{r^2}{2\beta'^2}\right)\tag{115}$$

$$\overset{x:=r^2}{\iff}-\frac{C_1\lambda_1}{C_2\lambda_2}\cdot\frac{2k+\frac{x}{\sigma'^2}}{2k+\frac{x}{\beta'^2}}=\exp\left(\frac{x}{2\sigma'^2}-\frac{x}{2\beta'^2}\right)\tag{116}$$

$$\iff-\frac{C_2\lambda_2}{C_1\lambda_1}\cdot\frac{2k+\frac{x}{\beta'^2}}{2k+\frac{x}{\sigma'^2}}=\exp\left(\frac{x}{2\beta'^2}-\frac{x}{2\sigma'^2}\right).\tag{117}$$

We focus on Eqn. (117). Without loss of generality, we assume $\sigma' > \beta'$, then both function $x \mapsto \frac{2k+x/\beta'^2}{2k+x/\sigma'^2}$ and function $x \mapsto \exp\left(x/(2\beta'^2) - x/(2\sigma'^2)\right)$ are monotonically increasing. If $\lambda_1\lambda_2 > 0$, the LHS and RHS of Eqn. (117) are continuous and monotonic in opposite directions. Thus, there is at most one solution to Eqn. (117). If $\lambda_1\lambda_2 < 0$, the LHS is monotonic increasing but the derivative is decreasing because

$$-\frac{C_2\lambda_2}{C_1\lambda_1} \cdot \frac{2k + \frac{x}{\beta'^2}}{2k + \frac{x}{\sigma'^2}} = -\frac{C_2\lambda_2}{C_1\lambda_1} \cdot \frac{1 + \frac{x}{2k\beta'^2}}{1 + \frac{x}{2k\sigma'^2}} \tag{118}$$

where the numerator $1 + \frac{x}{2k\beta'^2}$ is linearly increasing and the denominator $1 + \frac{x}{2k\sigma'^2}$ is also linearly increasing. On the other hand, the RHS is monotonic increasing and the derivative is also increasing because

$$\frac{1}{2\beta'^2} - \frac{1}{2\sigma'^2} > 0. \tag{119}$$

As a result, the difference function between RHS and LHS is monotone and there is at most one solution. Thus, we have shown Eqn. (117) has at most one solution.

Given that $(\lambda_1 g + \lambda_2 h)'$ is also continuous, we thus know the function $(\lambda_1 g + \lambda_2 h)$ is unimodal. □

Moreover, since g and h approach 0 when $r \to \infty$, $(\lambda_1 g + \lambda_2 h)$ approaches 0 when $r \to \infty$.

We define

$$\begin{aligned}&\overline{(\lambda_1 g + \lambda_2 h)^{-1}}(y) := \max y' \\ &\quad \text{s.t.} \quad \lambda_1 g(y') + \lambda_2 h(y') = x,\end{aligned} \tag{120}$$

$$\begin{aligned}&\underline{(\lambda_1 g + \lambda_2 h)^{-1}}(y) := \min y' \\ &\quad \text{s.t.} \quad \lambda_1 g(y') + \lambda_2 h(y') = x.\end{aligned} \tag{121}$$

Then, Lemma H.2 and $(\lambda_1 g + \lambda_2 h) \to 0$ when $r \to \infty$ imply that, when $y > 0$,

$$\begin{aligned}&y_0 \in \left(\underline{(\lambda_1 g + \lambda_2 h)^{-1}}(y),\ \overline{(\lambda_1 g + \lambda_2 h)^{-1}}(y)\right) \\ \iff &\lambda_1 g(y_0) + \lambda_2 h(y_0) > y.\end{aligned} \tag{122}$$

We simplify Eqn. (109) by observing that $g(\sigma'\sqrt{2t}) > 0$:

$$\begin{aligned}&\text{Eqn. (109)} \\ \iff &\sqrt{2t\sigma'^2 + r^2 + 2x_1 r} \\ &\quad \in \left(\underline{(\lambda_1 g + \lambda_2 h)^{-1}}(g(\sigma'\sqrt{2t})),\right. \\ &\qquad \left.\overline{(\lambda_1 g + \lambda_2 h)^{-1}}(g(\sigma'\sqrt{2t}))\right)\end{aligned} \tag{123}$$

$$\begin{aligned}\iff &\frac{\underline{(\lambda_1 g + \lambda_2 h)^{-1}}(g(\sigma'\sqrt{2t}))^2 - 2t\sigma'^2 - r^2}{2r} \le x_1 \\ &\le \frac{\overline{(\lambda_1 g + \lambda_2 h)^{-1}}(g(\sigma'\sqrt{2t}))^2 - 2t\sigma'^2 - r^2}{2r}.\end{aligned} \tag{124}$$

Combining Lemma H.1 with Eqn. (124) we get

$$\begin{aligned}&\Pr\left[\lambda_1 p(\boldsymbol{x}+\boldsymbol{\delta}) + \lambda_2 q(\boldsymbol{x}+\boldsymbol{\delta}) > g(\sigma'\sqrt{2t}) \,\middle|\, \|\boldsymbol{x}\|_2 = \sigma'\sqrt{2t}\right] \\ =&\mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{1}{2} + \frac{\overline{(\lambda_1 g + \lambda_2 h)^{-1}}(g(\sigma'\sqrt{2t}))^2 - 2t\sigma'^2 - r^2}{4r\sigma'\sqrt{2t}}\right) \\ &- \mathrm{BetaCDF}_{\frac{d-1}{2}}\left(\frac{1}{2} + \frac{\underline{(\lambda_1 g + \lambda_2 h)^{-1}}(g(\sigma'\sqrt{2t}))^2 - 2t\sigma'^2 - r^2}{4r\sigma'\sqrt{2t}}\right).\end{aligned} \tag{125}$$

Combining the above equation with (9.) yields the expression shown in Table 3. □

Proof of Theorem 10. To prove the theorem, the main work we need to do is deriving the closed-form expression for the inverse function g^{-1}, where

$$g(r) = \frac{1}{(2\sigma'^2)^{\frac{d}{2}-k}\pi^{\frac{d}{2}}} \cdot \frac{\Gamma(\frac{d}{2})}{\Gamma(\frac{d}{2}-k)} r^{-2k} \exp\left(-\frac{r^2}{2\sigma^2}\right). \tag{126}$$

(I.) When $k = 0$,

$$\begin{aligned}&\text{Eqn. (126)} \\ \iff &y = \frac{1}{(2\sigma'^2\pi)^{\frac{d}{2}}} \exp\left(-\frac{g^{-1}(y)^2}{2\sigma'^2}\right)\end{aligned} \tag{127}$$

$$\iff g^{-1}(y)^2 = -2\sigma'^2 \ln\left((2\sigma'^2\pi)^{\frac{d}{2}} y\right). \tag{128}$$

(II.) When $k > 0$, we notice that the Lambert W function W is the inverse function of $w \mapsto we^w$, i.e., $W(x)\exp(W(x)) = x$. We let the normalizing coefficient of $g(r)$ be

$$C := \frac{1}{(2\sigma'^2)^{\frac{d}{2}-k}\pi^{\frac{d}{2}}} \cdot \frac{\Gamma(\frac{d}{2})}{\Gamma(\frac{d}{2}-k)}. \tag{129}$$

Then

$$\begin{aligned}&\text{Eqn. (126)} \\ \iff &y = C g^{-1}(y)^{-2k} \exp\left(-\frac{g^{-1}(y)^2}{2\sigma'^2}\right)\end{aligned} \tag{130}$$

$$\iff \frac{C}{y} = g^{-1}(y)^2 k \exp\left(\frac{g^{-1}(y)^2}{2\sigma'^2}\right) \tag{131}$$

$$\iff \left(\frac{C}{y}\right)^{\frac{1}{k}} = g^{-1}(y)^2 \exp\left(\frac{g^{-1}(y)^2}{k\sigma'^2}\right) \tag{132}$$

$$\iff \frac{1}{2k\sigma^2}\left(\frac{C}{y}\right)^{\frac{1}{k}} = W^{-1}\left(\frac{g^{-1}(y)^2}{2k\sigma'^2}\right) \tag{133}$$

$$\iff g^{-1}(y)^2 = 2\sigma'^2 k W\left(\frac{1}{2k\sigma'^2}\left(\frac{C}{y}\right)^{\frac{1}{k}}\right). \tag{134}$$

Plugging in Eqns. (128) and (134) to $g^{-1}(\lambda_1 g(\sigma'\sqrt{2x}) + \lambda_2 h(\sigma'\sqrt{2x}))^2$ and $g^{-1}(\lambda_1 g(\beta'\sqrt{2x}) + \lambda_2 h(\beta'\sqrt{2x}))^2$ for $k = 0$ and $k > 0$ case, then results in Table 4 follow from algebra. □

Proof of Theorem 11. We prove the theorem by contradiction. Suppose that the $(\lambda_1,\, \lambda_2)$ that satisfy the constraint of Eqn. (12) are not unique, and we let $(\lambda_1^a,\, \lambda_2^a)$ and $(\lambda_1^b,\, \lambda_2^b)$ be such two pairs. Without loss of generality, we assume $\lambda_1^a \neq \lambda_1^b$.

If $\lambda_2^a = \lambda_2^b$, we have $P(\lambda_1^a,\, \lambda_2^a) = P(\lambda_1^b,\, \lambda_2^b)$, i.e., the region

$$\{\boldsymbol{x}-\boldsymbol{\delta} :\, p(\boldsymbol{x}-\boldsymbol{\delta}) \in \big[\min\{\lambda_1^a,\, \lambda_1^b\}p(\boldsymbol{x}) + \lambda_2^a q(\boldsymbol{x}),\, \max\{\lambda_1^a,\, \lambda_1^b\}p(\boldsymbol{x}) + \lambda_2^a q(\boldsymbol{x})\big)\} \tag{135}$$

has zero mass under distribution $\mathcal{P}$. Given that $\mathcal{P}$ and $\mathcal{Q}$ have positive and continuous density functions almost everywhere, the volume of the region is non-zero thus the mass under distribution $\mathcal{P}$ is also non-zero. Therefore, we also have $\lambda_2^a \neq \lambda_2^b$. Because of the partial monotonicity of P and Q functions (shown in Section 5.3), without loss of generality, we assume

$$\lambda_1^a < \lambda_1^b, \quad \lambda_2^a > \lambda_2^b. \tag{136}$$

Lemma H.3. *There exists a point $r_0 \geq 0$, either (1) or (2) is satisfied.*

(1) When $r > r_0$,

$$\begin{aligned}&\Pr[p(\boldsymbol{\epsilon}-\boldsymbol{\delta}) < \lambda_1^a p(\boldsymbol{\epsilon}) + \lambda_2^a q(\boldsymbol{\epsilon}) \,\big|\, \|\boldsymbol{\epsilon}\|_p = r]\\ \geq\, &\Pr[p(\boldsymbol{\epsilon}-\boldsymbol{\delta}) < \lambda_1^b p(\boldsymbol{\epsilon}) + \lambda_2^b q(\boldsymbol{\epsilon}) \,\big|\, \|\boldsymbol{\epsilon}\|_p = r];\end{aligned}$$

when $r < r_0$,

$$\begin{aligned}&\Pr[p(\boldsymbol{\epsilon}-\boldsymbol{\delta}) < \lambda_1^a p(\boldsymbol{\epsilon}) + \lambda_2^a q(\boldsymbol{\epsilon}) \,\big|\, \|\boldsymbol{\epsilon}\|_p = r]\\ \leq\, &\Pr[p(\boldsymbol{\epsilon}-\boldsymbol{\delta}) < \lambda_1^b p(\boldsymbol{\epsilon}) + \lambda_2^b q(\boldsymbol{\epsilon}) \,\big|\, \|\boldsymbol{\epsilon}\|_p = r].\end{aligned}$$

(2) When $r > r_0$,

$$\begin{aligned}&\Pr[p(\boldsymbol{\epsilon}-\boldsymbol{\delta}) < \lambda_1^a p(\boldsymbol{\epsilon}) + \lambda_2^a q(\boldsymbol{\epsilon}) \,\big|\, \|\boldsymbol{\epsilon}\|_p = r]\\ \leq\, &\Pr[p(\boldsymbol{\epsilon}-\boldsymbol{\delta}) < \lambda_1^b p(\boldsymbol{\epsilon}) + \lambda_2^b q(\boldsymbol{\epsilon}) \,\big|\, \|\boldsymbol{\epsilon}\|_p = r];\end{aligned}$$

when $r < r_0$,

$$\begin{aligned}&\Pr[p(\boldsymbol{\epsilon}-\boldsymbol{\delta}) < \lambda_1^a p(\boldsymbol{\epsilon}) + \lambda_2^a q(\boldsymbol{\epsilon}) \,\big|\, \|\boldsymbol{\epsilon}\|_p = r]\\ \geq\, &\Pr[p(\boldsymbol{\epsilon}-\boldsymbol{\delta}) < \lambda_1^b p(\boldsymbol{\epsilon}) + \lambda_2^b q(\boldsymbol{\epsilon}) \,\big|\, \|\boldsymbol{\epsilon}\|_p = r].\end{aligned}$$

Note that in $\Pr[\cdot \,\big|\, \|\boldsymbol{\epsilon}\|_p = r]$, the vector $\varepsilon \in \mathbb{R}^d$ is uniformly sampled from the ℓ_p-sphere of radius r.

Proof of Lemma H.3. For a given r, since $\mathcal{P}$ and $\mathcal{Q}$ are both ℓ_p-spherically symmetric, and the density functions are both positive almost everywhere, as long as

$$\lambda_1^a g(r) + \lambda_2^a h(r) \leq \lambda_1^b g(r) + \lambda_2^b h(r), \tag{137}$$

then

$$\begin{aligned}&[p(\boldsymbol{\epsilon}-\boldsymbol{\delta}) < \lambda_1^a p(\boldsymbol{\epsilon}) + \lambda_2^a q(\boldsymbol{\epsilon}) \,\big|\, \|\boldsymbol{\epsilon}\|_p = r]\\ \leq &[p(\boldsymbol{\epsilon}-\boldsymbol{\delta}) < \lambda_1^b p(\boldsymbol{\epsilon}) + \lambda_2^b q(\boldsymbol{\epsilon}) \,\big|\, \|\boldsymbol{\epsilon}\|_p = r].\end{aligned} \tag{138}$$

It still holds if we change the "$\leq$"'s to "$\geq$"'s in both Eqns. (137) and (138). Meanwhile,

$$\lambda_1^a g(r) + \lambda_2^a h(r) \leq \lambda_1^b g(r) + \lambda_2^b h(r) \iff \frac{g(r)}{h(r)} > \frac{\lambda_2^b - \lambda_2^a}{\lambda_1^a - \lambda_1^b}. \tag{139}$$

Since $g(r)/h(r)$ is strictly monotonic, there exists *at most* one point $r_0 \geq 0$ that divides

$$\frac{g(r)}{h(r)} > \frac{\lambda_2^b - \lambda_2^a}{\lambda_1^a - \lambda_1^b} \quad \text{and} \quad \frac{g(r)}{h(r)} < \frac{\lambda_2^b - \lambda_2^a}{\lambda_1^a - \lambda_1^b}. \tag{140}$$

If that r_0 exists, from Eqns. (137) to (139) the lemma statement follows.

Now we only need to show that r_0 exists. Assume that the point r_0 does not exist, it implies that for all r, we have either

$$\lambda_1^a g(r) + \lambda_2^a h(r) < \lambda_1^b g(r) + \lambda_2^b h(r) \tag{141}$$

or

$$\lambda_1^a g(r) + \lambda_2^a h(r) > \lambda_1^b g(r) + \lambda_2^b h(r) \tag{142}$$

while $P(\lambda_1^a,\, \lambda_2^a) = P(\lambda_1^b,\, \lambda_2^b) > 0$ and $Q(\lambda_1^a,\, \lambda_2^a) = Q(\lambda_1^b,\, \lambda_2^b) > 0$. It implies that for almost every r,

$$\Pr[p(\boldsymbol{\epsilon}-\boldsymbol{\delta}) \in (a,\, b) \,\big|\, \|\boldsymbol{\epsilon}\|_p = r] = 0 \tag{143}$$

where

$$\begin{aligned}a &= \min\{\lambda_1^a g(r) + \lambda_2^a h(r),\, \lambda_1^b g(r) + \lambda_2^b h(r)\},\\ b &= \max\{\lambda_1^a g(r) + \lambda_2^a h(r),\, \lambda_1^b g(r) + \lambda_2^b h(r)\}.\end{aligned} \tag{144}$$

This violates the continuous assumption on both $\mathcal{P}$ and $\mathcal{Q}$. Therefore, point r_0 exists. □

With Lemma H.3, we define auxiliary function $D : \mathbb{R}_+ \to \mathbb{R}$ such that

$$\begin{aligned}D(r) = &\Pr[p(\boldsymbol{\epsilon}-\boldsymbol{\delta}) < \lambda_1^a p(\boldsymbol{\epsilon}) + \lambda_2^a q(\boldsymbol{\epsilon}) \,|\, \|\boldsymbol{\epsilon}\|_p = r]\\ &- \Pr[p(\boldsymbol{\epsilon}-\boldsymbol{\delta}) < \lambda_1^b p(\boldsymbol{\epsilon}) + \lambda_2^b q(\boldsymbol{\epsilon}) \,|\, \|\boldsymbol{\epsilon}\|_p = r].\end{aligned} \tag{145}$$

We let $S(r)$ be the surface area of ℓ_p sphere of radius r. Then the P and Q can be written in integral form:

$$\begin{aligned}P(\lambda_1,\, \lambda_2) = &\int_0^\infty \Pr[p(\varepsilon-\boldsymbol{\delta}) < \lambda_1 p(\varepsilon) + \lambda_2 p(\varepsilon) \,|\\ &\|\varepsilon\|_p = r] \cdot g(r) S(r) \mathrm{d}r,\end{aligned} \tag{146}$$

$$Q(\lambda_1,\, \lambda_2) = \int_0^\infty \Pr[p(\varepsilon-\boldsymbol{\delta}) < \lambda_1 p(\varepsilon) + \lambda_2 p(\varepsilon) \,|$$

$$\|\varepsilon\|_p = r] \cdot h(r)S(r)\mathrm{d}r. \tag{147}$$

Since $P(\lambda_1^a, \lambda_2^a) = P(\lambda_1^b, \lambda_2^b)$ and $Q(\lambda_1^a, \lambda_2^a) = Q(\lambda_1^b, \lambda_2^b)$ by our assumption, simple arrangement yields

$$\int_0^{r_0} D(r)g(r)S(r)\mathrm{d}r = \int_{r_0}^{\infty} (-D(r))g(r)S(r)\mathrm{d}r \neq 0, \tag{148}$$

$$\int_0^{r_0} D(r)h(r)S(r)\mathrm{d}r = \int_{r_0}^{\infty} (-D(r))h(r)S(r)\mathrm{d}r \neq 0. \tag{149}$$

As Lemma H.3 shows, $D(r)$ where $r \in [0, r_0]$ always has the same sign as $-D(r)$ where $r \in [r_0, +\infty]$, and the last inequality ($\neq 0$) is again due to the continuity of p and q and the fact that $P(\lambda_1^a, \lambda_2^a) > 0$ and $Q(\lambda_1^a, \lambda_2^a) > 0$. Now we can divide Eqn. (148) by Eqn. (149) and apply the Cauchy's mean value theorem, which yields

$$\frac{D(\xi_1)g(\xi_1)S(\xi_1)}{D(\xi_1)h(\xi_1)S(\xi_1)} = \frac{D(\xi_2)g(\xi_2)S(\xi_2)}{D(\xi_2)h(\xi_2)S(\xi_2)}, \tag{150}$$

where $\xi_1 \in (0, r_0)$ and $\xi_2 \in (r_0, +\infty)$. Apparently, it requires

$$\frac{g(\xi_1)}{h(\xi_1)} = \frac{g(\xi_2)}{h(\xi_2)}. \tag{151}$$

However, g/h is strictly monotonic. By contradiction, there is no distinct pair $(\lambda_1^a, \lambda_2^a)$ and $(\lambda_1^b, \lambda_2^b)$ satisfying the constraint of Eqn. (12) simultaneously. □

Remark. Suppose $\mathcal{P}$ and $\mathcal{Q}$ are ℓ_p-radial extended Gaussian/Laplace distributions, i.e., their density functions $p(\boldsymbol{x})$ and $q(\boldsymbol{x})$ are

$$\begin{aligned} p(\boldsymbol{x}) &\propto \|\boldsymbol{x}\|_p^{-k} \exp\left(-\|\boldsymbol{x}\|_p/\sigma\right)^{\alpha}, \\ q(\boldsymbol{x}) &\propto \|\boldsymbol{x}\|_p^{-k} \exp\left(-\|\boldsymbol{x}\|_p/\beta\right)^{\alpha} \end{aligned} \tag{152}$$

with $\alpha > 0$ (for Gaussian $\alpha = 2$ and for Laplace $\alpha = 1$). Note that this is a broader family than the family considered in DSRS shown in the main text. we have

$$\frac{g}{h} \propto \exp(r/\beta - r/\sigma)^{\alpha} \tag{153}$$

that is strictly monotonic. Thus, Theorem 11 is applicable for this large family of smoothing distributions that are commonly used in the literature (Lécuyer et al., 2019; Cohen et al., 2019; Yang et al., 2020; Zhang et al., 2020a; Zhai et al., 2020; Jeong & Shin, 2020).

H.3. Discussion on Certification of Other ℓ_p Norms

DSRS can also be extended to provide a certified robust radius under ℓ_p norm other than ℓ_2.

Different from the case of ℓ_2 certification, for other ℓ_p norm the challenge is to compute $P(\lambda_1, \lambda_2)$, $Q(\lambda_1, \lambda_2)$, and $R(\lambda_1, \lambda_2)$ as defined in Theorem 5. For ℓ_2 certification, as shown by Theorems 5 and 9, there exist closed-form expressions for these quantities that can be efficiently implemented with numerical integrations. However, for other ℓ_p norms, finding such closed-form expressions for P, Q, and R is challenging.

Luckily, we notice that P, Q, and R are all probability-based definitions, as long as we can effectively sample from $\mathcal{P}$ and $\mathcal{Q}$ and effectively compute the density functions $p(\cdot)$ and $q(\cdot)$, we can estimate these function quantities from the empirical means of Monte-Carlo sampling.

Compared to numerical integration based ℓ_2 certification, Monte-Carlo sampling has *sampling uncertainty* and *efficiency* problems. Here we discuss these two problems in detail and how we can alleviate them.

Sampling Uncertainty and Mitigations. The empirical means for P and Q are stochastic, which breaks the nice properties of P and Q (shown in Section 5.3) with respect to (λ_1, λ_2) as the different queries to P and Q fluctuate around the actual value. Therefore, the joint binary search (Alg. 4) may fail to return the correct (λ_1, λ_2) pair. Thus, we propose a stabilization trick: the *same* set of samples is used when querying P and Q during the joint binary search. With this same set of samples, it can be easily verified that the nice properties (Propositions 1 and 2) still hold even if P and Q are empirical means. To guarantee the certification soundness in the context of probabilistic Monte-Carlo sampling, we introduce a different set of samples to test whether the solved (λ_1, λ_2) indeed upper bounds the intersection point (if the test fails we fall back to the classical Neyman-Pearson-based certification though it seldom happens). Since the test is also probabilistic, we need to accumulate this additional failing probability and use the lower bound of the confidence interval for soundness, which results in $1 - 2\alpha = 99.8\%$ certification instead of $1 - \alpha = 99.9\%$ as in classical randomized smoothing certification (Cohen et al., 2019; Yang et al., 2020). Note that the existence of additional confidence intervals caused by Monte-Carlo sampling makes DSRS based on Monte-Carlo sampling slightly looser than DSRS based on numerical integration.

Efficiency Concern and Mitigations. Traditionally we need to sample several (denoted as M, in our implementation we set $M = 5 \times 10^4$) high-dimensional vectors for *each* P or Q computation, which induces the efficiency concern. With the usage of the aforementioned stabilization trick, now we only sample one set of M samples during the whole joint binary search instead of during each P and Q computation. Combining with the testing phase sampling, the whole algorithm needs to sample $2M$ vectors rather than $\mathcal{O}(M \log^2 M)$ without the stabilization trick, so

it greatly solves the efficiency concern. Moreover, we notice that for the samples $\{\epsilon_i\}_{i=1}^M$ we only need to care about its densities $\{p(\varepsilon_i)\}_{i=1}^M$, $\{q(\varepsilon_i)\}_{i=1}^M$ and $\{p(\varepsilon_i - \boldsymbol{\delta})\}_{i=1}^M$. Thus, we store only these three quantities instead of the whole d-dimensional vectors $\{\epsilon_i\}_{i=1}^M$, reducing the space complexity from $\mathcal{O}(M \times d)$ to $\mathcal{O}(M)$.

These techniques significantly improve efficiency in practice. Although DSRS based on Monte-Carlo sampling is still slightly looser and slower than DSRS based on numerical integration, DSRS based on Monte-Carlo sampling makes certifying robustness under other ℓ_p norms feasible. In this work, we focus on the ℓ_2 norm, because additive randomized smoothing is not optimal for other norms (e.g., ℓ_1 (Levine & Feizi, 2021)) or the state-of-the-art certification can be directly translated from ℓ_2 certification (e.g., ℓ_∞ (Yang et al., 2020) and semantic transformations (Li et al., 2021)). Moreover, to the best of our knowledge, standard ℓ_2 certification is the most challenging setting where additive randomized smoothing achieves state-of-the-art and no other work can achieve visibly tighter robustness certification than standard Neyman-Pearson certification (Yang et al., 2020; Levine et al., 2020; Mohapatra et al., 2020).

I. Implementation and Optimizations

In this appendix, we discuss the implementation tricks and optimizations, along with our simple heuristic for selecting the hyperparameter T in the additional smoothing distribution $\mathcal{Q} = \mathcal{N}^{\mathsf{g}}_{\mathsf{trunc}}(k, T, \sigma)$.

I.1. Implementation Details

We implement DSRS in Python with about two thousand lines of code. The tool uses `PyTorch`[2] to query a given base classifier with Monte-Carlo sampling in order to derive the confidence intervals $[\underline{P_A}, \overline{P_A}]$ and $[\underline{Q_A}, \overline{Q_A}]$. Then, the tool builds the whole DSRS pipeline on `SciPy`[3] and `NumPy`[4]. Specifically, the numerical integration is implemented with `scipy.integrate.quad()` method. We exploit the full independence across the certification for different input instances and build the tool to be fully parallelizable on CPUs. By default, we utilize 10 processes, and the number of processes can be dynamically adjusted.

The tool is built in a flexible way that adding new smoothing distributions is not only theoretically straightforward but also easy in practice. In the future, we plan to extend the tool to 1) provide GPU support; 2) reuse existing certification results from previous instances with similar confidence intervals to achieve higher efficiency. We will also support more smoothing distributions.

[2] http://pytorch.org/

[3] https://scipy.org/scipylib/index.html

[4] https://numpy.org/

In the implementation, we widely use the logarithmic scale, since many quantities in the computation have varied scales. For example, since the input dimension d is typically over 500 on a real-world dataset, the density functions p and q decay very quickly along with the increase of noise magnitude. Another example is the input variable for $\ln(\cdot)$ and $W(\cdot)$ in Theorem 5. These variables are exponential with respect to the input dimension d so they could be very large or small. To mitigate this, we perform all computations with varied scales in logarithmic scale to improve the precision and floating-point soundness. For example, we implement a method `lnlogadd` to compute $\log(\lambda_1 \exp(x_1) + \lambda_2 \exp(x_2))$ and apply method `wlog` in (Yang et al., 2020) to compute $W(\exp(x))$. We remark that in the binary search for dual variables (see Section 5.3), we also use the logarithmic scale for λ_1 and λ_2.

The code, model weights, and all experiment data are publicly available at https://github.com/llylly/DSRS.

I.2. T Heuristics

As briefly outlined in Section 6.1, we apply a simple yet effective heuristic to determine the hyperparameter T in additional smoothing distribution $\mathcal{Q} = \mathcal{N}^{\mathsf{g}}_{\mathsf{trunc}}(k, T, \sigma)$.

Concretely, we first sample the prediction probability from the original smoothing distribution $\mathcal{P}$ and get the confidence lower bound $\underline{P_A}$ of $P_A = f^{\mathcal{P}}(\boldsymbol{x}_0)_{y_0}$. Then, we use the following empirical expression to determine T from $\underline{P_A}$:

$$T = \sigma \sqrt{\frac{2d}{d-2k} \Gamma\mathrm{CDF}^{-1}_{d/2-k}(p)}, \tag{154}$$

where

$$p = \max\{-0.08 \ln(1 - \underline{P_A}) + 0.2, 0.5\}. \tag{155}$$

It can be viewed as we first parameterize T by p and then find a simple heuristic to determine p by $\underline{P_A}$.

The T's parameterization with p is inspired by the probability mass under original smoothing distribution $\mathcal{P} = \mathcal{N}^{\mathsf{g}}(k, \sigma)$ if true-decision region is concentrated in a ℓ_2-ball centered at $\boldsymbol{x}_0$. Concretely, from $\mathcal{P}$'s definition,

$$\Pr_{\boldsymbol{\epsilon} \sim \mathcal{P}}[\|\boldsymbol{\epsilon}\|_2 \le T] = p. \tag{156}$$

Then, we use a randomly sampled CIFAR-10 training set containing $1,000$ points with models trained using Consistency (Jeong & Shin, 2020) under $\sigma = 0.50$ to sweep all $p \in \{0.1, 0.2, \cdots, 0.9\}$. We plot the minimum p and maximum p that gives the highest certified robust radius as a segment and find Eqn. (155) fits the general tendency well as shown in Appendix I.2. Thus, we use this simple heuristic to determine T. Empirically, this simple heuristic generalizes well and is competitive with more complex methods as shown in Appendix J.

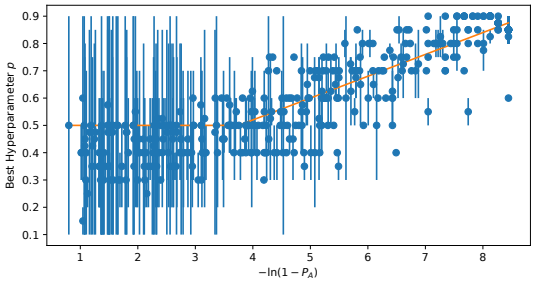


Figure 7. Our ρ heuristic (Eqn. (155)) shown as orange curve fits the generalization tendency of optimal ρ ranges (shown as blue segments) well.

Another heuristic that we have deployed is the fall-back strategy. When the empirical probability $\widetilde{P_A} = 1$, i.e., if for all the sampled $\boldsymbol{\epsilon} \sim \mathcal{P}$, $F(\boldsymbol{x}_0 + \boldsymbol{\epsilon}) = y_0$, then we fall back to still using $\mathcal{P}$ instead of using another distribution $\mathcal{Q}$ for the second round of sampling. This strategy is inspired by a finding that, with the fixed sampling budget, if $\underline{P_A}$ is already very high, it is more efficient to use more samples to further increase the confidence interval of $\underline{P_A}$ rather than querying imprecise information under another distribution $\mathcal{Q}$. Notice that such strategy does not break the high-confidence soundness of our certification, because $\Pr[\widetilde{P_A} = n/N \,|\, P_A \leq t] \geq \Pr[\widetilde{P_A} = n/N \wedge \widetilde{P_A^{\text{half}}} = 1 \,|\, P_A \leq t]$ for any $t < 1$ and Bernoulli distributed sampling (which is our case), and we use Bernoulli confidence interval that corresponds to $\Pr[\widetilde{P_A} = n/N \,|\, P_A \leq t]$. Due to the tight experiment time, we only deployed this strategy on ImageNet evaluation but not on MNIST and CIFAR-10 evaluations.

I.3. Training Strategy for Generalized Gaussian Smoothing

We train the base classifiers on each dataset using Gaussian augmentation training (Cohen et al., 2019), Consistency training (Jeong & Shin, 2020), and SmoothMix training (Jeong et al., 2021), which are typical training methods for randomized smoothing. We do not consider other training approaches such as (Zhai et al., 2020; Salman et al., 2019; Li et al., 2019a; Carmon et al., 2019) because: (1) Some training approaches either require additional data (Carmon et al., 2019; Salman et al., 2019), or relatively time-consuming (Salman et al., 2019), or are reported to be not as effective as later approaches (Li et al., 2019a); (2) Selected training approaches are widely used or achieve state-of-the-art with high training efficiency.

On MNIST, for all training methods, we use a convolutional neural network with 4 convolutional layers and 3 fully connected layers following Cohen et al. (2019) as the base classifiers' architecture. On CIFAR-10, for all training methods, we use ResNet-110 (He et al., 2016) as the base classifiers' architecture. These architecture settings follow the prior work on smoothed classifier training (Cohen et al., 2019; Salman et al., 2019; Zhai et al., 2020).

On both MNIST and CIFAR-10, for all training methods, we train for 150 epochs. The learning rate is 0.01 and is decayed by 0.1 at the 50th and 100th epoch. For Consistency training, the hyperparameter $\lambda = 5$ on MNIST and $\lambda = 20$ on CIFAR-10. We use two noise vectors per training batch per instance. These are the best hyperparameters reported in (Jeong & Shin, 2020). The batch size is 256 on both MNIST and CIFAR-10 following (Jeong & Shin, 2020). For SmoothMix training, we directly use the best hyperparameters from their open-source repository: `https://github.com/jh-jeong/smoothmix/blob/main/EXPERIMENTS.MD`.

On ImageNet, we use ResNet-50 (He et al., 2016) as the base classifiers' architecture and finetune from the open-source model trained by Cohen et al. (Cohen et al., 2019) with Gaussian smoothing. We train for 10 epochs due to the expensive training cost on ImageNet and we remark that better results can be achieved with a larger training time budget. The learning rate is 0.001 and is decayed at the end of every epoch by 0.1. For Consistency training, the hyperparameter $\lambda = 5$ and we use two noise vectors per training batch per instance following the best hyperparameters reported in (Jeong & Shin, 2020). For SmoothMix training, since the open-source repository does not contain the best hyperparameters, we select the hyperparameters as suggested in the original paper (Jeong et al., 2021).

During training, the training samples are augmented by adding noises following *training smoothing distribution*. In typical training approaches (Cohen et al., 2019; Carmon et al., 2019; Salman et al., 2019; Zhai et al., 2020; Jeong & Shin, 2020), the training smoothing distribution is set to be the same as the original smoothing distribution $\mathcal{P}$ for constructing the smoothed classifier. However, for our generalized Gaussian $\mathcal{N}^{\text{g}}(k, \sigma)$ as $\mathcal{P}$ with large k, we find this strategy gives a poor empirical performance.

To better train the base classifier for our original smoothing distribution $\mathcal{P}$ with large k, we introduce a warm-up stage on the training smoothing distribution. Suppose our original smoothing distribution $\mathcal{P}$ for constructing the smoothed classifier is $\mathcal{N}^{\text{g}}(k_0, \sigma)$ We let $e_0 = 100$ be the number of warm-up epochs on MNIST and CIFAR-10, or $e_0 = 10000$ be the number of warm-up steps on ImageNet. In the first e_0 epochs (on MNIST and CIFAR-10) or steps (on ImageNet), we use the training smoothing distribution with smaller k. Formally, in the eth epoch/step where $e \leq e_0$, the training

smoothing distribution $\mathcal{P}' = \mathcal{N}^{\mathtt{g}}(k, \sigma)$ where

$$k = \lceil k_0 - k_0^{1-e/e_0} \rceil. \tag{157}$$

For later epochs/steps, we use the original smoothing distribution $\mathcal{P}$ itself as the training smoothing distribution. This strategy gradually increases the k of training smoothing distribution throughout the training, so that the base classifier can be a better fit for the final desired distribution $\mathcal{P}$. Using this strategy, the smoothed classifier constructed from our trained base classifier has similar certified robustness compared to standard Gaussian augmentation under classical Neyman-Pearson-based certification.

J. Additional Experimental Results

In this appendix, we present additional experiment results and studies.

J.1. Empirical Study Setup of Concentration Property

In Figure 4 in Appendix B, we present our investigation of the decision regions of base classifiers. The investigation follows the following protocol: (1) We choose the base classifier from (Salman et al., 2019) on ImageNet trained for $\sigma = 0.5$ Gaussian smoothing as the subject. The reason is that this base classifier is one of the state-of-the-art certifiably robust classifiers on the large-scale ImageNet dataset and our code uses the same preprocessing parameters so it is easy to adopt their model. (2) We pick every 500-th image from the test set of the ImageNet dataset to form a subset of 100 samples. (3) We filter out the samples where the base classifier cannot classify correctly even without adding any noise, which results in 89 remaining samples. (4) For each of these 89 samples, for each perturbation magnitude r, we uniformly sample 1000 perturbation vectors from the hypersphere with ℓ_2-radius r and compute the empirical probability of true-prediction, where the step size of r is 10.

Figure 4 implies that for a vast majority of samples, the true-prediction samples are highly concentrated on an ℓ_2 ball around the clean input since there exists apparent ℓ_2 magnitude thresholds where the true-prediction probability is almost 1 within the thresholds and almost 0 beyond the thresholds. This implies that the concentration property may be achievable for real-world base classifiers in randomized smoothing.

In Figure 9, we follow the same protocol but plot the landscape of base classifiers trained using generalized Gaussian distribution (instead of standard Gaussian distribution as in Figure 4). By comparing Figure 9 and Figure 4, we find that although base classifiers in Figure 4 can achieve better certified robustness using Neyman-Pearson certification and generalized Gaussian smoothing (compare Figure 2(b) and Neyman-Pearson rows in Table 10), they also sacrifice the concentration property, which can explain why DSRS improvements are much smaller on models in Section 6 compared with models in Figure 2(b). Thus, as discussed in Section 6, there may be a large space for exploring training approaches that favor DSRS certification by preserving the concentration property.

J.2. Effectiveness of T-Heuristics and Attempts on Better Optimization Tricks

Better Optimization Tricks. For obtaining a tighter certified radius, we should make the support of $\mathcal{N}^{\mathtt{g}}_{\mathsf{trunc}}(k, T, \sigma)$ more aligned with the decision region while keeping the Q_A large enough. So except using a simple heuristic, we can also turn the search for an appropriate value of T into an optimization problem. In order to make the optimization more stable, here we will construct the optimization based on P_{con} that is more scale-invariant, and then transform it to get the final appropriate $T = \sigma\sqrt{2\Gamma\mathrm{CDF}^{-1}_{d/2}(P_{\mathsf{con}})}$.

The final optimization objective is now built as $P_{\mathsf{con}} = \arg\min -Q_A + \frac{\lambda}{2}(P_{\mathsf{con}} - P_A)^2$, where λ is a hyperparameter that controls the relative weight of the two loss terms. The Q_A here is estimated by sampling from the distribution $\mathcal{N}^{\mathtt{g}}_{\mathsf{trunc}}(k, T, \sigma)$ in which the T is determined by P_{con}; however, the actual process of such sampling is implemented by rejecting the sampled noise whose norm is bigger than T. Therefore, there will be no gradient obtained for P_{con} through the backward of the loss, namely, the optimized objective. So instead, we attempt to approximate the gradient comes from the first loss term $-Q_A$ with $G_{Q_A} = \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{Q}}[\nabla_{||\boldsymbol{\epsilon}||_2} \mathit{CrossEntropyLoss}(f(\boldsymbol{x}_0 + \boldsymbol{\epsilon}), y_0)]$. Then, we will only have the gradient information, and there is no explicit form of Q_A anymore. The P_A is estimated with the $\underline{P_A}$ which we have already obtained, so the final estimation of the gradient for P_{con} is $G_{Q_A} + \lambda(P_{\mathsf{con}} - \underline{P_A})$.

Experiment Setting. The P_{con} is optimized for different input test images respectively, and the initialized P_{con} is set to 0.7. For each input, we will update P_{con} 20 steps on datasets MNIST and CIFAR10 while updating only 10 steps on dataset ImageNet to reduce time complexity. For each step, we will sample $2,000$ times for estimating the term G_{Q_A}, and the learning rate is set to $2,000$. Besides, to avoid the P_{con} being optimized too large or too small, we will clip the final optimized P_{con} within 0.1 and 0.9. Since the optimization is a bit time-consuming, we only test it on CIFAR10 with $\sigma = 0.5$ and test it on MNIST and ImageNet with $\sigma = 1.0$. Different λ is also tried for different datasets and different training methods for getting better performance.

Performance. The final results are shown in Table 5, and the certification approach based on the optimization tricks

mentioned above is denoted as "Opt" in the table. As we can see, our simple T-Heuristics could still be competitive with the method based on complicated optimization but with a cheaper cost.

J.3. Full Curves and Separated Tables

J.3.1. Separate Tables by Smoothing Variance

Due to the page limit, in the main text (Section 6, Table 2), we aggregate the certified robust accuracy across models with different smoothing variance $\sigma \in \{0.25, 0.50, 1.00\}$. To show the full landscape, we present the certified robust accuracy for each model trained with each variance. The evaluation protocol is the same as the one in the main text, and the tables for MNIST, CIFAR-10, and ImageNet models are Table 6, Table 7, and Table 8 respectively. We observe that DSRS outperforms standard Neyman-Pearson certification for a wide range of radii.

J.3.2. Curves

Following the convention (Cohen et al., 2019; Salman et al., 2019), we plot the certified robust accuracy - radius curve in Figure 8.

The curves correspond to the certified robust accuracy data in Table 2 (in Section 6), i.e., the certified robust accuracy under each radius r is the maximum certified robust accuracy among models trained with variance $\sigma \in \{0.25, 0.50, 1.00\}$. We observe that among all medium to large radii (including those not shown in Table 2), DSRS provides higher certified robust accuracy than Neyman-Pearson certification. The margin of DSRS is relatively small on CIFAR-10 but is apparent on MNIST and ImageNet. Especially, the margins on ImageNet reflect that DSRS is particularly effective on large datasets.

J.3.3. ACR Results

In the literature, another common metric of certified robustness is ACR (average certified radius) (Zhai et al., 2020; Jeong & Shin, 2020; Jeong et al., 2021). In Table 9, we report the ACR comparison between Neyman-Pearson-based certification and our DSRS certification. Across the three smoothing variance choices $\sigma \in \{0.25, 0.50, 1.00\}$, we find $\sigma = 1.00$ yields the highest ACR, so we only report the ACR for models smoothed with $\sigma = 1.00$. As we can see, in all cases, DSRS significantly improves over Neyman-Pearson-based certification in terms of ACR.

J.4. Using Distribution with Different Variance as $\mathcal{Q}$

We take the models trained with Gaussian augmentation (Cohen et al., 2019) and variance $\sigma = 1.00$ as examples. We use "DSRS-trunc" to represent DSRS using truncated generalized Gaussian as $\mathcal{Q}$, and "DSRS-var" to represent DSRS using generalized Gaussian with different variance as $\mathcal{Q}$, and compare their robustness certification (i.e., certified robust accuracy) in Table 10. From the table, we find that on MNIST and CIFAR-10, DSRS-var is better than DSRS-trunc, whereas on ImageNet, DSRS-trunc is slightly better than DSRS-var. Both DSRS-trunc and DSRS-var are significantly better than Neyman-Pearson-based certification.

To investigate the reason, we follow the protocols for studying the concentration property in Appendix J.1 to plot the landscape of models on MNIST, CIFAR-10, and ImageNet, as shown in Figure 9. From the figure, we find that the curves on ImageNet are generally steeper, which corresponds to that the concentration property is better satisfied on ImageNet. Therefore, we conjecture that when the concentration property (see Definition 3) is better satisfied, DSRS with truncated Gaussian as $\mathcal{Q}$ is better than Gaussian with different variance as $\mathcal{Q}$.

J.5. Comparison with Higher-Order Randomized Smoothing

It is difficult to have a direct comparison with higher-order randomized smoothing (Mohapatra et al., 2020; Levine et al., 2020), which is the only work to the best of our knowledge that uses additional information beyond P_A to tighten the robustness certification in randomized smoothing. This difficulty comes from the following reasons: (1) Higher-order randomized smoothing only supports standard Gaussian smoothing, while DSRS is particularly useful with generalized Gaussian smoothing. (2) All experiment evaluations in higher-order randomized smoothing are conducted with large sampling numbers ($N = 2 \times 10^5$ on CIFAR-10 and $N = 1.25 \times 10^6$ on ImageNet) that makes the evaluation costly, while DSRS is designed for practical sampling numbers ($N = 10^5$). (3) The code is not open-source yet (Mohapatra et al., 2020). Nonetheless, we can directly compare with the curves provided by Mohapatra et al. (2020).

We capture the certified robust accuracy vs. ℓ_2 radius r curves from (Mohapatra et al., 2020) in Figure 10. As we can see, compared with Neyman-Pearson-based certification, the improvements from higher-order randomized smoothing are small especially on the large ImageNet dataset despite the excessive sampling numbers (1.25×10^6). In contrast, as shown in Figure 8, within only 10^5 sampling number, DSRS is visibly tighter than Neyman-Pearson-based certification. In fact, to the best of our knowledge, DSRS is the first model-agnostic approach that is visibly tighter than Neyman-Pearson-based certification under ℓ_2 radius.

Table 5. ℓ_2 certified robust accuracy w.r.t. different radii r for different certification approaches.

Dataset	Training Method	Certification Approach	Certified Accuracy under Radius r											
			0.2	0.4	0.6	0.8	1.0	1.2	1.4	1.6	1.8	2.0	2.2	2.4
MNIST	Gaussian Aug. ($\sigma = 1.00$)	Neyman-Pearson	95.5 %	93.5 %	90.0 %	86.1 %	80.4 %	72.8 %	61.4 %	50.2 %	36.6 %	25.2 %	14.5 %	8.5 %
		DSRS(T-heuristic)	95.5 %	93.5 %	90.2 %	86.9 %	81.4 %	74.4 %	64.6 %	55.2 %	42.8 %	30.9 %	20.3 %	11.3 %
		Opt ($\lambda = 7e-05$)	95.5%	93.5%	90.0%	86.9%	81.7%	74.9%	65.6%	55.8%	43.8%	30.5%	19.1%	10.2%
	Consistency ($\sigma = 1.00$)	Neyman-Pearson	94.5 %	92.6 %	89.3 %	85.9 %	80.7 %	74.4 %	65.9 %	56.9 %	44.1 %	34.4 %	23.3 %	12.8 %
		DSRS(T-heuristic)	94.5 %	92.8 %	89.3 %	86.3 %	81.4 %	76.1 %	68.3 %	59.5 %	50.7 %	39.8 %	30.7 %	20.0 %
		Opt ($\lambda = 6e-06$)	94.5%	92.8%	89.3%	86.2%	81.2%	75.8%	68.2%	59.6%	50.5%	39.6%	30.7%	19.8%
			Certified Accuracy under Radius r											
			0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0	1.1	1.2
CIFAR-10	Gaussian Aug. ($\sigma = 0.50$)	Neyman-Pearson	60.4 %	55.2 %	51.3 %	45.9 %	40.8 %	35.6 %	30.1 %	24.3 %	20.0 %	16.7 %	13.0 %	10.1 %
		DSRS(T-heuristic)	60.6 %	55.5 %	51.5 %	46.8 %	42.1 %	37.3 %	32.5 %	27.4 %	22.8 %	19.3 %	16.0 %	12.7 %
		Opt ($\lambda = 3e-06$)	60.6%	55.5%	51.3%	46.7%	42.0%	37.2%	32.5%	27.4%	23.0%	19.3%	16.0%	12.5%
	Consistency ($\sigma = 0.50$)	Neyman-Pearson	53.1 %	50.5 %	48.6 %	45.5 %	43.6 %	41.5 %	38.7 %	36.7 %	35.1 %	32.0 %	29.1 %	25.7 %
		DSRS(T-heuristic)	53.1 %	50.7 %	48.7 %	45.7 %	44.0 %	41.8 %	39.6 %	37.8 %	36.0 %	34.4 %	31.3 %	28.6 %
		Opt ($\lambda = 4e-06$)	53.1%	50.7%	48.7%	45.7%	44.0%	41.8%	39.5%	37.8%	36.0%	34.4%	31.4%	28.4%
ImageNet	Gaussian Aug. ($\sigma = 1.00$)	Neyman-Pearson	57.5 %	55.1 %	52.2 %	49.7 %	47.0 %	43.9 %	40.8 %	38.1 %	35.0 %	33.2 %	29.6 %	25.3 %
		DSRS(T-heuristic)	57.7 %	55.6 %	52.7 %	51.0 %	48.4 %	45.5 %	43.1 %	40.2 %	37.9 %	35.3 %	32.8 %	30.5 %
		Opt ($\lambda = 1e-05$)	57.7%	55.5%	52.4%	50.5%	48.2%	45.0%	42.9%	40.0%	38.0%	35.0%	32.7%	29.9%
	Consistency ($\sigma = 1.00$)	Neyman-Pearson	55.9 %	54.4 %	53.0 %	51.2 %	48.2 %	46.2 %	44.2 %	41.7 %	39.1 %	36.4 %	34.4 %	32.1 %
		DSRS(T-heuristic)	56.0 %	54.6 %	53.1 %	51.8 %	49.9 %	47.4 %	45.7 %	44.2 %	41.7 %	39.3 %	37.8 %	35.8 %
		Opt ($\lambda = 1e-05$)	56.0%	54.6%	53.1%	51.8%	49.7%	47.4%	45.3%	44.0%	41.6%	39.3%	37.8%	35.9%

Table 6. MNIST: Certified robust accuracy for models smoothed with different variance σ certified with different certification approaches.

Variance	Training Method	Certification Approach	Certified Accuracy under Radius r											
			0.25	0.50	0.75	1.00	1.25	1.50	1.75	2.00	2.25	2.50	2.75	3.00
0.25	Gaussian Aug. (Cohen et al., 2019)	Neyman-Pearson	97.9%	96.4%	92.1%									
		DSRS	97.9%	96.6%	92.7%									
	Consistency (Jeong & Shin, 2020)	Neyman-Pearson	98.4%	97.5%	94.4%									
		DSRS	98.4%	97.5%	95.4%									
	SmoothMix (Jeong et al., 2021)	Neyman-Pearson	98.6%	97.6%	96.5%									
		DSRS	98.6%	97.7%	96.8%									
0.50	Gaussian Aug. (Cohen et al., 2019)	Neyman-Pearson	97.8%	96.9%	94.6%	88.4%	78.7%	52.6%						
		DSRS	97.8%	97.0%	95.0%	89.8%	83.4%	59.1%						
	Consistency (Jeong & Shin, 2020)	Neyman-Pearson	98.4%	97.3%	96.0%	92.3%	83.8%	67.5%						
		DSRS	98.4%	97.3%	96.0%	93.5%	87.1%	71.8%						
	SmoothMix (Jeong et al., 2021)	Neyman-Pearson	98.2%	97.1%	95.4%	91.9%	85.1%	73.0%						
		DSRS	98.1%	97.1%	95.9%	93.4%	87.5%	76.6%						
1.00	Gaussian Aug. (Cohen et al., 2019)	Neyman-Pearson	95.2%	91.9%	87.7%	80.6%	71.2%	57.6%	41.0%	25.5%	13.6%	6.2%	2.1%	0.9%
		DSRS	95.1%	91.8%	88.2%	81.5%	73.6%	61.6%	48.4%	34.1%	21.0%	10.6%	4.4%	1.2%
	Consistency (Jeong & Shin, 2020)	Neyman-Pearson	93.9%	90.9%	86.4%	80.8%	73.0%	61.1%	49.1%	35.6%	21.7%	10.4%	4.1%	1.9%
		DSRS	93.9%	91.1%	86.9%	81.7%	75.2%	65.6%	55.8%	41.9%	31.4%	17.8%	8.6%	2.8%
	SmoothMix (Jeong et al., 2021)	Neyman-Pearson	92.0%	88.9%	84.4%	78.6%	69.8%	60.7%	49.9%	40.2%	31.5%	22.2%	12.2%	4.9%
		DSRS	92.2%	89.0%	84.8%	79.7%	72.0%	63.9%	54.4%	46.2%	37.6%	29.2%	18.5%	7.2%

K. Extended Related Work Discussion

In this appendix, we discuss related work from two branches: training approaches for randomized smoothing, and data-dependent randomized smoothing. Both branches aim to improve the certified robustness of randomized smoothing. For tighter certification approaches leveraging additional information, a detailed discussion can be found in Appendix L.1. For more related work we refer the interested readers to recent surveys and books (Liu et al., 2021; Li et al., 2020b; Albarghouthi, 2021).

To improve the certified robustness of randomized smoothing, efforts have been made on both the training (Li et al., 2019a; Zhai et al., 2020; Salman et al., 2019; Jeong & Shin, 2020) and the certification sides (Lécuyer et al., 2019; Cohen et al., 2019; Li et al., 2019a; Yang et al., 2020; Zhang et al., 2020a; Yang et al., 2022). On the training side, data augmentation (Cohen et al., 2019), regularization (Li et al., 2019a; Zhai et al., 2020; Jeong & Shin, 2020), and adversarial training (Salman et al., 2019) help to train stable base models under noise corruptions so that higher certified robustness for a smoothed classifier can be achieved. In this work, we focus on certification, and these training approaches can be used in conjunction with ours to provide higher certified robustness.

Another potential way to improve the certified robustness of randomized smoothing is to dynamically change the smoothing distribution $\mathcal{P}$ based on the input toward maximizing the certified radius (Alfarra et al., 2020; Eiras et al., 2021; Schuchardt et al., 2022). In this scenario, the certification needs to take into account that the attacker may adaptively mislead the pipeline to choose a "bad" smoothing distri-

Table 7. CIFAR-10: Certified robust accuracy for models smoothed with different variance σ certified with different certification approaches.

Variance	Training Method	Certification Approach	Certified Accuracy under Radius r 0.25	0.50	0.75	1.00	1.25	1.50	1.75	2.00	2.25	2.50	2.75	3.00
0.25	Gaussian Aug. (Cohen et al., 2019)	Neyman-Pearson	56.1%	35.7%	13.4%									
		DSRS	57.4%	39.4%	17.3%									
	Consistency (Jeong & Shin, 2020)	Neyman-Pearson	61.8%	50.9%	34.7%									
		DSRS	62.5%	52.5%	37.8%									
	SmoothMix (Jeong et al., 2021)	Neyman-Pearson	63.9%	53.3%	38.4%									
		DSRS	64.7%	55.5%	41.1%									
0.50	Gaussian Aug. (Cohen et al., 2019)	Neyman-Pearson	53.7%	41.3%	27.7%	17.1%	9.1%	2.8%						
		DSRS	54.1%	42.7%	30.6%	20.3%	12.6%	4.0%						
	Consistency (Jeong & Shin, 2020)	Neyman-Pearson	49.2%	43.9%	38.0%	32.3%	23.8%	18.1%						
		DSRS	49.6%	44.1%	38.7%	35.2%	28.1%	19.7%						
	SmoothMix (Jeong et al., 2021)	Neyman-Pearson	53.2%	47.6%	40.2%	34.2%	26.7%	19.6%						
		DSRS	53.3%	48.5%	42.1%	35.9%	29.4%	21.7%						
1.00	Gaussian Aug. (Cohen et al., 2019)	Neyman-Pearson	40.2%	32.6%	24.7%	18.9%	14.9%	10.2%	7.5%	4.1%	2.0%	0.7%	0.1%	0.1%
		DSRS	40.3%	33.1%	25.9%	20.6%	16.1%	12.5%	8.4%	6.4%	3.5%	1.8%	0.7%	0.1%
	Consistency (Jeong & Shin, 2020)	Neyman-Pearson	37.2%	32.6%	29.6%	25.9%	22.5%	19.0%	16.4%	13.8%	11.2%	9.0%	7.1%	5.1%
		DSRS	37.1%	32.5%	29.8%	27.1%	23.5%	20.9%	17.6%	15.3%	13.1%	10.9%	8.9%	6.5%
	SmoothMix (Jeong et al., 2021)	Neyman-Pearson	43.2%	39.5%	33.9%	29.1%	24.0%	20.4%	17.0%	13.9%	10.3%	7.8%	4.9%	2.3%
		DSRS	43.2%	39.7%	34.9%	30.0%	25.8%	22.1%	18.7%	16.1%	13.2%	10.2%	7.1%	3.9%

Table 8. ImageNet: Certified robust accuracy for models smoothed with different variance σ certified with different certification approaches.

Variance	Training Method	Certification Approach	Certified Accuracy under Radius r 0.25	0.50	0.75	1.00	1.25	1.50	1.75	2.00	2.25	2.50	2.75	3.00
0.25	Gaussian Aug. (Cohen et al., 2019)	Neyman-Pearson	57.1%	41.6%	17.4%									
		DSRS	58.4%	47.9%	24.4%									
	Consistency (Jeong & Shin, 2020)	Neyman-Pearson	59.8%	49.8%	36.9%									
		DSRS	60.4%	52.4%	40.4%									
	SmoothMix (Jeong et al., 2021)	Neyman-Pearson	46.7%	38.2%	28.2%									
		DSRS	47.4%	40.0%	29.8%									
0.50	Gaussian Aug. (Cohen et al., 2019)	Neyman-Pearson	53.6%	48.3%	43.3%	36.8%	31.4%	24.5%						
		DSRS	53.7%	49.9%	44.7%	39.3%	34.8%	27.4%						
	Consistency (Jeong & Shin, 2020)	Neyman-Pearson	53.6%	48.3%	43.3%	36.8%	31.4%	24.5%						
		DSRS	53.7%	49.9%	44.7%	39.3%	34.8%	27.4%						
	SmoothMix (Jeong et al., 2021)	Neyman-Pearson	38.7%	33.5%	28.8%	24.6%	18.1%	13.5%						
		DSRS	39.1%	34.9%	30.3%	26.8%	21.6%	15.6%						
1.00	Gaussian Aug. (Cohen et al., 2019)	Neyman-Pearson	42.5%	37.2%	33.0%	29.2%	24.8%	21.4%	17.6%	13.7%	10.2%	7.8%	5.7%	3.6%
		DSRS	42.5%	38.1%	34.4%	30.2%	27.0%	23.3%	21.3%	18.7%	14.2%	11.0%	9.0%	5.7%
	Consistency (Jeong & Shin, 2020)	Neyman-Pearson	40.0%	38.3%	34.2%	31.8%	28.7%	25.6%	22.1%	19.1%	16.1%	14.0%	10.6%	8.5%
		DSRS	40.2%	38.5%	35.4%	32.6%	30.7%	28.1%	25.4%	22.6%	19.6%	17.4%	14.1%	10.4%
	SmoothMix (Jeong et al., 2021)	Neyman-Pearson	29.8%	25.6%	21.8%	19.2%	17.0%	14.2%	11.8%	10.1%	8.9%	7.2%	6.0%	4.6%
		DSRS	29.7%	26.2%	23.0%	20.6%	18.0%	15.7%	14.0%	12.1%	9.9%	8.4%	7.2%	5.3%

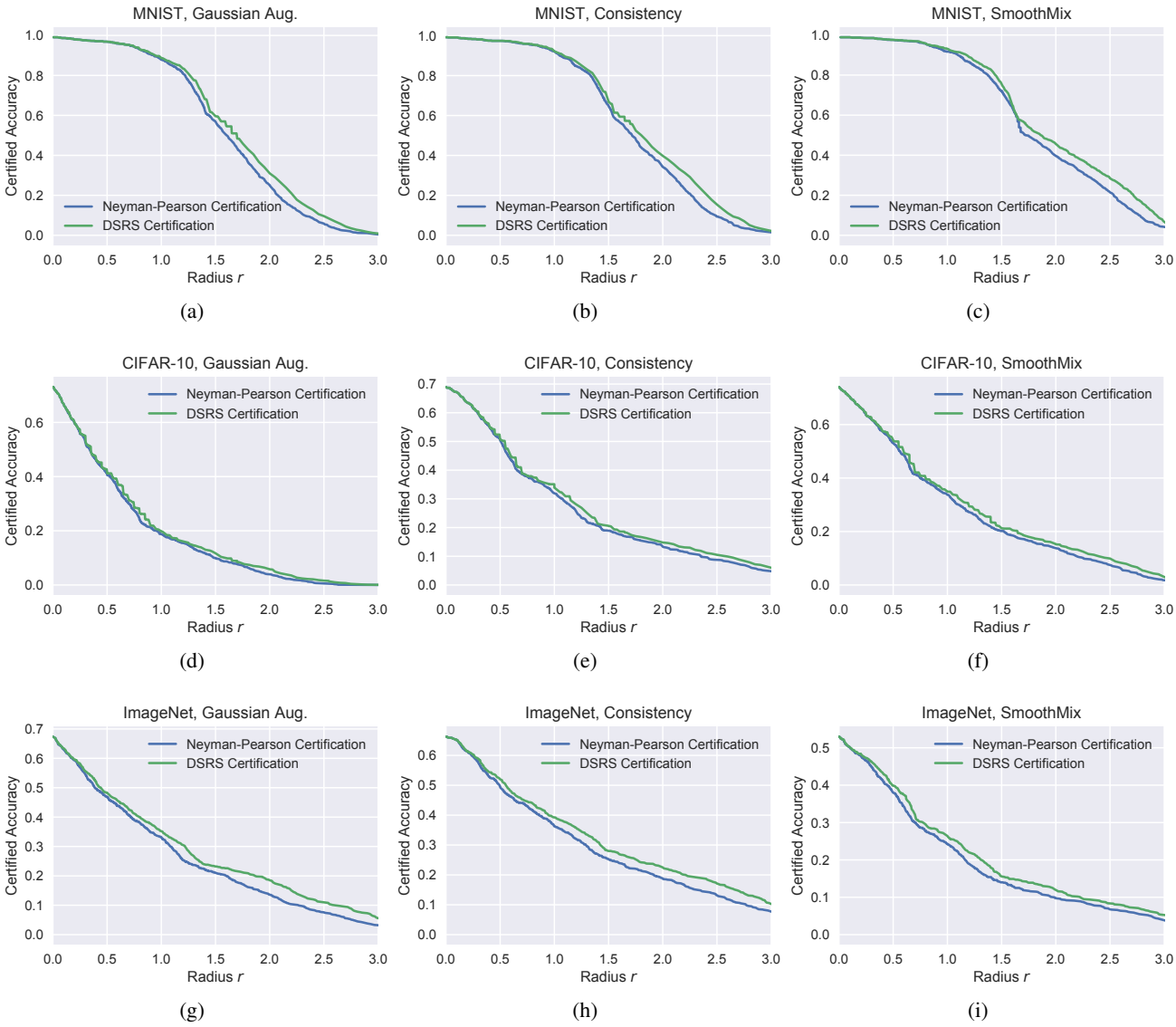
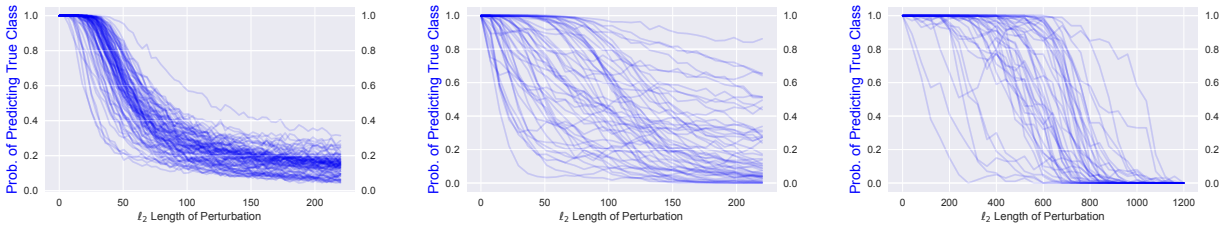


Figure 8. Certified robust accuracy - radius curve corresponding to Table 2.

Table 9. Average certified radius (ACR) statistics. The smoothing variance hyperparameter $\sigma = 1.00$. The evaluation protocol is the same as that in the main text.

Training Method	Certification	MNIST	CIFAR-10	ImageNet
Gaussian Augmentation	Neyman-Pearson	1.550	0.447	0.677
	DSRS	1.629	0.469	0.750
	Relative Improvement	**+5.10%**	**+4.92%**	**+10.78%**
Consistency	Neyman-Pearson	1.645	0.636	0.796
	DSRS	1.730	0.659	0.862
	Relative Improvement	**+5.17%**	**+3.62%**	**+8.29%**
SmoothMix	Neyman-Pearson	1.716	0.676	0.490
	DSRS	1.806	0.712	0.525
	Relative Improvement	**+5.24%**	**+5.33%**	**+7.14%**

bution. Therefore, additional costs such as memorizing training data need to be paid to defend such adaptive robustness vulnerabilities. A recent work (Súkeník et al., 2021) shows that input-dependent randomized smoothing may not bring substantial improvements in certified robustness. In DSRS, we select the additional smoothing distribution $\mathcal{Q}$ dynamically based on the input, which may appear like input-dependent randomized smoothing. However, we select such distribution $\mathcal{Q}$ only for certification purposes, and the original distribution $\mathcal{P}$ that is used to construct the smoothed classifier remains static. Thus, we do not need to consider the existence of adaptive attackers. Indeed, for any smoothing distribution $\mathcal{Q}$, with DSRS, we generate valid robustness certification for the static smoothed classifier $\widetilde{F}^{\mathcal{P}}$.



(a) MNIST model. (b) CIFAR-10 model. (c) ImageNet model.

Figure 9. Probability of true-prediction w.r.t. ℓ_2 length of perturbations for base classifiers from Gaussian augmentation training studied in Appendix J.4. Figures are plotted following the same protocol as in Appendix J.1.

Table 10. Comparison of DSRS certified robust accuracy with different types of additional smoothing distribution $\mathcal{Q}$ and Neyman-Pearson-based certification. Detail experiment settings are in Appendix J.4.

Dataset	Certification	Certified Accuracy under Radius r											
		0.25	0.50	0.75	1.00	1.25	1.50	1.75	2.00	2.25	2.50	2.75	3.00
MNIST	Neyman-Pearson	**95.2%**	**91.9%**	87.7%	80.6%	71.2%	57.6%	41.0%	25.5%	13.6%	6.2%	2.1%	0.9%
	DSRS-trunc	95.1%	91.8%	87.9%	81.3%	72.9%	60.2%	46.1%	30.9%	17.4%	9.4%	3.6%	**1.2%**
	DSRS-var	95.1%	91.8%	**88.2%**	**81.5%**	**73.6%**	**61.6%**	**48.4%**	**34.1%**	**21.0%**	**10.6%**	**4.4%**	**1.2%**
CIFAR-10	Neyman-Pearson	40.2%	32.6%	24.7%	18.9%	14.9%	10.2%	7.5%	4.1%	2.0%	0.7%	0.1%	**0.1%**
	DSRS-trunc	**40.3%**	32.9%	25.5%	20.1%	15.7%	11.5%	8.0%	5.5%	2.7%	1.5%	0.6%	**0.1%**
	DSRS-var	**40.3%**	**33.1%**	**25.9%**	**20.6%**	**16.1%**	**12.5%**	**8.4%**	**6.4%**	**3.5%**	**1.8%**	**0.7%**	**0.1%**
ImageNet	Neyman-Pearson	42.5%	37.2%	33.0%	29.2%	24.8%	21.4%	17.6%	13.7%	10.2%	7.8%	5.7%	3.6%
	DSRS-trunc	42.5%	38.1%	34.4%	30.2%	**27.0%**	**23.3%**	**21.3%**	**18.7%**	14.2%	**11.0%**	**9.0%**	**5.7%**
	DSRS-var	**42.9%**	**38.5%**	**35.0%**	**31.0%**	26.5%	23.2%	21.0%	18.3%	**14.6%**	10.5%	8.2%	5.3%

L. Discussions on Generalizing DSRS Framework

In this appendix, we first introduce prior work that leverages additional information for certification in randomized smoothing, then generalize our DSRS as a more general framework to theoretically compare with the related work and highlight future directions.

L.1. Existing Work on Leveraging Additional Information for Certification

We discuss all known work that leverages more information to achieve tighter robustness certification for randomized smoothing prior to this paper to the best of our knowledge.

Additional Black-Box Information. Our DSRS leverages additional information to tighten the certification for randomized smoothing. We leverage the information from an additional smoothing distribution. This information can be obtained from the base classifier that we *only* have query access on the predicted label. We call the information from this limited query access "black-box" information. The certification approaches that only require black-box information can be applied to any classification model regardless of the model structure. Thus, they are usually more general and more scalable. The classical Neyman-Pearson certification only requires black-box information.

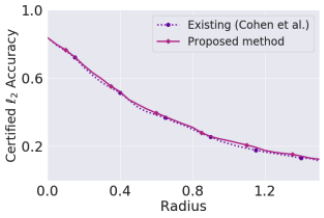
Besides our DSRS, the only other form of additional black-box information that is leveraged is the higher-order information (Levine et al., 2020; Mohapatra et al., 2020). Formally, our additional black-box information has the form

$$\Pr_{\boldsymbol{\epsilon}\sim\mathcal{Q}}[F_0(\boldsymbol{x}+\boldsymbol{\epsilon})=y]. \tag{158}$$

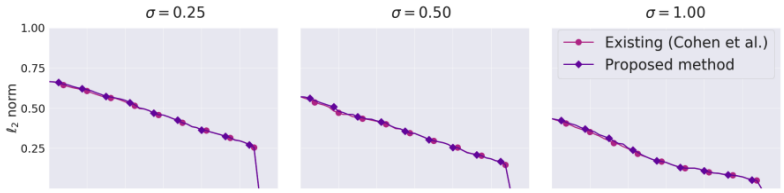
In contrast, the higher-order information, especially second-order information used by Levine et al. (2020); Mohapatra et al. (2020) has the form

$$\|\nabla f_0^{\mathcal{P}}(\boldsymbol{x}_0)_{y_0}\|_p = \|\nabla \Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}[F_0(\boldsymbol{x}_0+\boldsymbol{\epsilon})=y_0]\|_p \tag{159}$$

that is also shown to be estimable given the black-box query access. However, the higher-order information has several limitations: (1) It is hard to leverage the information beyond the second order. Therefore, only second-order information is used in existing certification approaches yet. However, to achieve optimal tightness, one needs to leverage infinite orders of information, which brings an infinite number of constraints and is thus intractable. In contrast, we show that extra information from only one additional distribution suffices to derive a strongly tight certification. (2) In practice, the second-order information shows marginal improvements in the widely used ℓ_2 and ℓ_∞ certification settings on real-world datasets (Levine et al., 2020; Mohapatra et al., 2020) and even such improvements require a large number of samples (usually in million order instead of ours 10^5).



(a) (Mohapatra et al., 2020, Figure 2(b)): Higher-order randomized smoothing on CIFAR-10.



(b) (Mohapatra et al., 2020, Figure 4): Higher-order randomized smoothing on ImageNet for models trained with different smoothing variances.

Figure 10. Higher-order randomized smoothing certification (solid curves) compared with standard Neyman-Pearson-based certification (dotted curves).

Dvijotham et al. (2020) also propose to use additional information to tighten the robustness certification for randomized smoothing ("full-information" setting). They formalize the tightest possible certification and compare it with Neyman-Pearson-based certifiation ("information-limited" setting), but in practice, they do not try to leverage information from distributions other than $\mathcal{P}$.

Constraining Model Structure. If we discard the "black-box information" constraint, we can obtain tighter robustness certification than classical Neyman-Pearson. For example, knowing the model structure can benefit the certification. Lee et al. (2019) show that when the base classifier is a decision tree, we can use dynamic programming to derive a strongly tight certification against ℓ_0-bounded perturbations. Awasthi et al. (2020) show that, if the base classifier first performs a known low-rank projection, then works on the low-rank projected space, for the corresponding smoothed classifier, we can have a tighter certification on both ℓ_2 and ℓ_∞ settings. However, it is challenging to find a projection such that the model preserves satisfactory performance while the projection rank is low. Indeed, the approach is evaluated on DCT basis to show the improvement on ℓ_∞ certification, and there exists a gap between the actual achieved certified robustness and the state of the art. We do not compare with these approaches since they impose additional assumptions on the base classifiers so their applicable scenarios are limited, and currently, the state-of-the-art base classifier does not satisfy their imposed constraints under ℓ_2 and ℓ_∞ certification settings. Recently, for ℓ_1 certification, a deterministic and improved smoothing approach (a type of non-additive smoothing mechanism) is proposed to handle the case where input images are constrained in space $\{0, 1/255, \cdots, 244/255, 1\}^d$ (Levine & Feizi, 2021). This could be viewed as constraining the attack space from another aspect and implies that certified robustness can be improved by better smoothing mechanisms, which is orthogonal to our work that focuses on certification for existing smoothing mechanisms.

Confidence Smoothing. A group of certification approaches assumes that the base classifier outputs normalized confidence on the given input, and the smoothed classifier predicts the class with the highest expectation of normalized confidence under noised input. This assumption can be viewed as a special type of "Constraining Model Structure". Under this assumption, we can query and approximate the *quantile* of the confidence under noised input: $F_0(\boldsymbol{x}_0 + \boldsymbol{\epsilon})$ where $\boldsymbol{\epsilon} \sim \mathcal{P}$. With this information, we can leverage the Neyman-Pearson lemma in a more delicate way to provide a tighter (higher) lower bound of the expected confidence under perturbation, i.e., $\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{P}} F_0(\boldsymbol{x} + \boldsymbol{\delta} + \boldsymbol{\epsilon})$.

These certification approaches provide tighter certification than the classical Neyman-Pearson for the smoothed classifier that predicts the class with the highest expected normalized confidence. They are also useful for regression tasks such as object detection in computer vision as shown in (Chiang et al., 2020). However, for the classification task, for utilizable base classifiers (i.e., benign accuracy $> 50\%$ under noise), if we simply set the predicted class to have 100% confidence, we only increase the certified radius of the classifier and the certification for this "one-hot" base classifier only requires classical Neyman-Pearson. Thus, these certification approaches, e.g., (Kumar et al., 2020a), may not achieve higher certified robustness on the classification task than Neyman-Pearson and therefore we do not compare with them.

L.2. General Framework

Focusing on the certification with additional black-box information, we generalize the DSRS to allow more general additional information.

Definition 5 (General Additional Black-Box Information)**.** For given base classifier F_0, suppose the true label at $\boldsymbol{x}_0$ is y_0, for certifying robustness at $\boldsymbol{x}_0$, the general additional

black-box information has the form

$$\begin{cases} \int_{\mathbb{R}^d} f_1(\boldsymbol{x})\mathbb{I}\left[F_0(\boldsymbol{x}) = y_0\right] \mathrm{d}\boldsymbol{x} = c_1, \\ \cdots \\ \int_{\mathbb{R}^d} f_i(\boldsymbol{x})\mathbb{I}\left[F_0(\boldsymbol{x}) = y_0\right] \mathrm{d}\boldsymbol{x} = c_i, \\ \cdots \\ \int_{\mathbb{R}^d} f_N(\boldsymbol{x})\mathbb{I}\left[F_0(\boldsymbol{x}) = y_0\right] \mathrm{d}\boldsymbol{x} = c_N, \end{cases} \tag{160}$$

where f_i and c_i are pre-determined; f_i is integrable in $\mathbb{R}^d$ and $c_i \in \mathbb{R}$ for $1 \leq i \leq N$.

Remark. Obtaining the information in Eqn. (160) requires only the black-box access to whether $F_0(\boldsymbol{x})$ equals to y_0. We define the general additional black-box information in this way because: (1) The information from finite points is useless since the smoothed classifier has zero probability mass on finite points, so the useful information is based on integration; (2) To provide a lower bound of $\widetilde{F}_0^{\mathcal{P}}(\boldsymbol{x}_0 + \boldsymbol{\delta})_{y_0}$, we only need to care whether $F_0(\boldsymbol{x})$ equals to y_0 in region of interest.

Examples. (1) Our DSRS, the additional information $\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{Q}}[f(\boldsymbol{\epsilon})] = Q_A$ instantiates Definition 5 by setting $N = 1$, $f_1(\cdot) = q(\cdot - \boldsymbol{x}_0)$ and $c_1 = Q_A$. (2) In (Mohapatra et al., 2020; Levine et al., 2020), the second-order information $\nabla f_0^{\mathcal{P}}(\boldsymbol{x}_0)$ instantiates Definition 5 by setting $N = d$, $f_i(\boldsymbol{x}) = -\nabla p(\boldsymbol{x} - \boldsymbol{x}_0)_i$, and $c_i = \left(\nabla f_0^{\mathcal{P}}(\boldsymbol{x}_0)\right)_i$ according to Theorem 1 in (Mohapatra et al., 2020). We remark that due to the sampling difficulty, instead of using the whole vector $\nabla f_0^{\mathcal{P}}(\boldsymbol{x}_0)$ as the information, second-order smoothing (Mohapatra et al., 2020; Levine et al., 2020) uses its p-norm in practice. However, using the full information only gives tighter certification so we consider this a more ideal variant.

Then, we can extend the constrained optimization problem ($\mathbf{C}$) in Section 5.1 to ($\mathbf{C}^{\text{ext}}$) to accommodate the general information as such

$$\begin{aligned} \underset{f}{\text{minimize}} \quad & \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{P}}[f(\boldsymbol{\delta} + \boldsymbol{\epsilon})] \\ \text{s.t.} \quad & \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{P}}[f(\boldsymbol{\epsilon})] = P_A, \\ & \int_{\mathbb{R}^d} f_1(\boldsymbol{\epsilon}) f(\boldsymbol{\epsilon}) \mathrm{d}\boldsymbol{\epsilon} = c_1, \\ & \cdots \\ & \int_{\mathbb{R}^d} f_N(\boldsymbol{\epsilon}) f(\boldsymbol{\epsilon}) \mathrm{d}\boldsymbol{\epsilon} = c_N, \\ & 0 \leq f(\boldsymbol{\epsilon}) \leq 1 \quad \forall \boldsymbol{\epsilon} \sim \mathbb{R}^d. \end{aligned} \tag{161}$$

Similarly, by the strong duality (Theorem 3), to solve the certification problem

$$\forall \boldsymbol{\delta} \text{ s.t. } \|\boldsymbol{\delta}\|_p \leq r,\ \mathbf{C}_{\boldsymbol{\delta}}^{\text{ext}}(P_A,\, c_1,\, \ldots,\, c_N) > 0.5, \tag{162}$$

we only need to solve the dual problem ($\mathbf{D}^{\text{ext}}$):

$$\underset{\eta,\lambda_1,\ldots,\lambda_N\in\mathbb{R}}{\text{maximize}} \ \Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}\left[p(\boldsymbol{\epsilon}) < \eta p(\boldsymbol{\epsilon} + \boldsymbol{\delta}) + \sum_{i=1}^N \lambda_i f_i(\boldsymbol{\epsilon} + \boldsymbol{\delta})\right] \tag{163}$$

$$\begin{aligned} \text{s.t.} \ & \Pr_{\boldsymbol{\epsilon}\sim\mathcal{P}}\left[p(\boldsymbol{\epsilon} - \boldsymbol{\delta}) < \eta p(\boldsymbol{\epsilon}) + \sum_{i=1}^N \lambda_i f_i(\boldsymbol{\epsilon})\right] = P_A, \\ & \int_{\mathbb{R}^d} \mathbb{I}\left[p(\boldsymbol{\epsilon} - \boldsymbol{\delta}) < \eta p(\boldsymbol{\epsilon}) + \sum_{i=1}^N \lambda_i f_i(\boldsymbol{\epsilon})\right] f_1(\boldsymbol{\epsilon}) \mathrm{d}\boldsymbol{\epsilon} = c_1, \\ & \cdots \\ & \int_{\mathbb{R}^d} \mathbb{I}\left[p(\boldsymbol{\epsilon} - \boldsymbol{\delta}) < \eta p(\boldsymbol{\epsilon}) + \sum_{i=1}^N \lambda_i f_i(\boldsymbol{\epsilon})\right] f_N(\boldsymbol{\epsilon}) \mathrm{d}\boldsymbol{\epsilon} = c_N. \end{aligned}$$

We remark that this generalization shares a similar spirit as one type of generalization of Neyman-Pearson Lemma (Chernoff & Scheffe, 1952; Mohapatra et al., 2020). Following the same motivation, Dvijotham et al. (2020) try to generalize the certification by adding more constraints in their "full-information setting". However, it is unclear whether their constraints in f-divergences form have the same expressive power as ours in practice (i.e., the practicality of theoretically tight Hockey-Stick divergence). A study of these different types of generalization would be our future work.

More importantly, we believe that the pipeline of DSRS can be adapted to solve this generalized dual problem. We hope that this generalization and the corresponding DSRS could enable the exploration and exploitation of more types of additional information for tightening the robustness certification of randomized smoothing.

Implications. The generalized DSRS framework allows us to explicitly compare different types of additional information. For example, comparing our additional distribution information and higher-order information, we find that (1) for additional distribution information, from Theorem 1 and Corollary 1, the strong tightness can be acheived for $N = C - 1$ where C is the number of classes; (2) for higher-order information, from (Mohapatra et al., 2020, Asymptotic-Optimality Remark), the strong tightness can be achieved when all orders of information are used, i.e., $N \to \infty$. This comparison suggests that our additional information from another smoothing distribution should be more efficient.

Another implication is that, from Corollary 1, under multiclass setting, with proper choices of the $(C-1)$ additional smoothing distributions, if we base DSRS on solving dual problem ($\mathbf{D}^{\text{ext}}$), the DSRS can achieve *strong tightness* in multiclass setting.

L.3. Limitations and Future Directions

Despite the promising theoretical and empirical results of DSRS, DSRS still has some limitations that open an avenue for future work. We list the following future directions: (1) tighter certification from a more ideal additional smoothing distribution $\mathcal{Q}$: there may exist better smoothing distribution $\mathcal{Q}$ or better methods to optimize hyperparameters in $\mathcal{Q}$ than what we have considered in this work in terms of certifying larger certified radius in practice; (2) better training approach for DSRS: there may be a large space for exploring training approaches that favor DSRS certification since all existing training methods are designed for Neyman-Pearson-based certification. We believe that advances in this aspect can boost the robustness certification with DSRS to achieve state-of-the-art certified robustness. (3) better additional information: more generally, besides the prediction probability from an additional smoothing distribution, there may exist other useful additional information for certification in randomized smoothing. We hope our generalization of the DSRS framework in this appendix can inspire future work in tighter and more efficient certification for randomized smoothing.