On the robustness of randomized classifiers to adversarial examples

Rafael Pinot, Laurent Meunier, Florian Yger, Cédric Gouy-Pailler, Yann Chevaleyre, Jamal Atif

On the robustness of randomized classifiers to adversarial examples

Adversarial generalization through noise injection

Rafael Pinot[*1] Laurent Meunier[*2,3] Florian Yger[2]
Cédric Gouy-Pailler[4] Yann Chevaleyre[2] Jamal Atif[2]

[1] Ecole Polytechnique Fédérale de Lausanne
[2] LAMSADE, Université Paris-Dauphine
[3] Facebook AI Research, Paris
[4] Institut LIST, CEA, Université Paris-Saclay

Abstract

This paper investigates the theory of robustness against adversarial attacks. We focus on randomized classifiers (*i.e.* classifiers that output random variables) and provide a thorough analysis of their behavior through the lens of statistical learning theory and information theory. To this aim, we introduce a new notion of robustness for randomized classifiers, enforcing local Lipschitzness using probability metrics. Equipped with this definition, we make two new contributions. The first one consists in devising a new upper bound on the adversarial generalization gap of randomized classifiers. More precisely, we devise bounds on the generalization gap and the adversarial gap (*i.e.* the gap between the risk and the worst-case risk under attack) of randomized classifiers. The second contribution presents a yet simple but efficient noise injection method to design robust randomized classifiers. We show that our results are applicable to a wide range of machine learning models under mild hypotheses. We further corroborate our findings with experimental results using deep neural networks on standard image datasets, namely CIFAR-10 and CIFAR-100. All robust models we trained models can simultaneously achieve state-of-the-art accuracy (over 0.82 clean accuracy on CIFAR-10) and enjoy *guaranteed* robust accuracy bounds (0.45 against ℓ_2 adversaries with magnitude 0.5 on CIFAR-10).

1 Introduction

In the last few years, there has been a growing concern on adversarial example attacks in machine learning. An adversarial attack refers to a small (humanly imperceptible) change of an input specifically designed to fool a machine learning model. These attacks have recently come to light thanks to works by [5] and [50] studying deep neural networks for image classification, although it was an existing topic in spam filter analysis [12, 17, 31]. The vulnerability of state-of-the-art classifiers to these attacks has genuine security implications especially for deep neural networks used in AI-driven technologies such as self-driving cars, as repetitively demonstrated by [46, 48] and [61]. Besides security issues, this shows how little we know about the worst-case behaviors of models the industry uses daily. It is essential for the community to understand the very nature of this phenomenon in order to mitigate the threat.

Accordingly, a large body of works has been trying to design new models that would be less vulnerable to the adversarial setting [18, 21, 33, 56, 58] but most of them were proven (in time) to offer only limited protection against more sophisticated attacks [1, 6, 11, 20, 51]. Among the defense strategies, randomization has proven effective in some contexts [13, 30, 40, 59]. Albeit these significant efforts, randomization techniques lack theoretical arguments. In this paper, we generalize the prior results from Pinot et al. [39] by studying a general class of randomized classifiers, including randomized neural networks, for which we demonstrate adversarial robustness guarantees and analyze their generalization properties.

1.1 Supervised learning for image classification in a nutshell

Let us consider the supervised classification problem with an input space $\mathcal{X}$ and an output space $\mathcal{Y}$. In the following, w.l.o.g. we will consider $\mathcal{X} \subset [-1,1]^d$ to be a set of images, and $\mathcal{Y} := [K] := \{1, \dots, K\}$ a set of labels describing them. The goal of a supervised machine learning algorithm is to design classifier that maps any image $\boldsymbol{x} \in \mathcal{X}$ to a label $y \in \mathcal{Y}$. To do so, the learner has access to a *training sample* of n image-label pairs $\mathcal{S} := \{(\boldsymbol{x_1}, y_1), \dots, (\boldsymbol{x_n}, y_n)\}$. Each training pair $(\boldsymbol{x_i}, y_i)$ is assumed to be drawn *i.i.d.* from a ground-truth distribution $\mathcal{D}$. To build a classifier, the usual strategy is to select a hypothesis function $\boldsymbol{h} : \mathcal{X} \to \mathcal{Y}$ from a pre-defined hypothesis class $\mathcal{H}$ to minimize the *risk* with respect to $\mathcal{D}$. This risk minimization problem writes

$$\inf_{\boldsymbol{h} \in \mathcal{H}} \mathcal{R}(\boldsymbol{h}) := \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}} \left[\mathcal{L}_{0/1} \left(\boldsymbol{h}(\boldsymbol{x}), y \right) \right] , \tag{1}$$

where $\mathcal{L}_{0/1}$ represents the 0/1 loss that outputs 1 when $\boldsymbol{h}(\boldsymbol{x}) \neq y$, and zero otherwise.

In practice, the learner does not have access to the ground-truth distribution; hence it cannot estimate the risk $\mathcal{R}(\boldsymbol{h})$. To find an approximate solution for Problem (1), a learning algorithm solves the *empirical risk minimization* problem instead. In this case, we simply replace the risk by its empirical counterpart over the training sample $\mathcal{S} := \{(\boldsymbol{x_1}, y_1), \dots, (\boldsymbol{x_n}, y_n)\}$. The empirical risk minimization problem writes

$$\inf_{\boldsymbol{h} \in \mathcal{H}} \mathcal{R}_{\mathcal{S}}(\boldsymbol{h}) := \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}_{0/1} \left(\boldsymbol{h}(\boldsymbol{x_i}), y_i \right) . \tag{2}$$

Then, to evaluate how far the selected hypothesis is from the optimum, one wants to upper bound the difference between the risk and the empirical risk of any $\boldsymbol{h} \in \mathcal{H}$. This difference is known as the *generalization gap*.

1.2 Classification in the presence of an adversary

Given a hypothesis $\boldsymbol{h} \in \mathcal{H}$ and a sample $(\boldsymbol{x}, y) \sim \mathcal{D}$, the goal of an adversary is to find a perturbation $\boldsymbol{\tau} \in \mathcal{X}$ such that the following assertions *both* hold. First, the perturbation is imperceptible to humans. This means that a human cannot visually distinguish the standard example $\boldsymbol{x}$ from the *adversarial example* $\boldsymbol{x} + \boldsymbol{\tau}$. Second, the perturbation modifies $\boldsymbol{x}$ enough to make the classifier misclassify. More formally, the adversary seeks a perturbation $\boldsymbol{\tau} \in \mathcal{X}$ such that $\boldsymbol{h}(\boldsymbol{x} + \boldsymbol{\tau}) \neq y$.

Although the notion of imperceptible modification is very natural for humans, it is genuinely hard to formalize. Despite these difficulties, in the image classification setting, a sufficient condition to ensure that the attack will remain undetected is to constrain the perturbation $\boldsymbol{\tau}$ to have a small ℓ_p norm. This means that for any $p \in [1, \infty]$, there exists a threshold $\alpha_p > 0$ for which any perturbation $\boldsymbol{\tau}$ is imperceptible as soon as $\|\boldsymbol{\tau}\|_p \leq \alpha_p$. The literature on adversarial attacks for image classification usually uses either an ℓ_∞ norm akin [32] or an ℓ_2 norm akin [6] as a surrogate for imperceptibility. Other authors such as [8] and [36] also used an ℓ_1 norm or an ℓ_0 semi-norm.

To account for adversaries possibly manipulating the input images, one needs to revisit the standard risk minimization by incorporating the adversary in the problem. The goal becomes to minimize the *worst-case* risk under α_p-bounded manipulations. We call this problem the *adversarial risk minimization*. It writes

$$\inf_{\boldsymbol{h} \in \mathcal{H}} \mathcal{R}^{\text{adv}}(\boldsymbol{h}; \alpha_p) := \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}} \left[\sup_{\boldsymbol{\tau} \in B_p(\alpha_p)} \mathcal{L}_{0/1} \left(\boldsymbol{h}(\boldsymbol{x} + \boldsymbol{\tau}), y \right) \right] , \tag{3}$$

where $B_p(\alpha_p) := \{\tau \in \mathcal{X} \ s.t. \ \|\boldsymbol{\tau}\|_p \leq \alpha_p\}$. In this new formulation, the adversary focuses on optimizing the inner maximization, while the learner tries to get the best hypothesis from $\mathcal{H}$ "under attack". By analogy with the standard setting, given n training examples $\mathcal{S} := \{(\boldsymbol{x_1}, y_1), \dots, (\boldsymbol{x_n}, y_n)\}$, we want to find an approximate solution to the adversarial risk minimization by studying its empirical counterpart, the *empirical adversarial risk minimization*. This optimization problem writes

$$\inf_{\boldsymbol{h} \in \mathcal{H}} \mathcal{R}^{\text{adv}}_{\mathcal{S}}(\boldsymbol{h}; \alpha_p) := \frac{1}{n} \sum_{i=1}^{n} \sup_{\boldsymbol{\tau} \in B_p(\alpha_p)} \mathcal{L}_{0/1} \left(\boldsymbol{h}(\boldsymbol{x_i} + \boldsymbol{\tau}), y_i \right) . \tag{4}$$

In the presence of an adversary, two major issues appear in the empirical risk minimization. First, as recently pointed out by [32], the adversarial generalization error (*i.e.* the gap between the empirical adversarial risk and the adversarial risk) can be much larger than in the standard setting. Indeed, the adversary makes the problem dependent on the dimension of $\mathcal{X}$. Hence, in high-dimension (*e.g.* for images) one needs much more samples to classify correctly as pointed out by [44] as well as [47]. Moreover, finding an approximate solution to the adversarial risk minimization is not always sufficient. Indeed, recent works by [52] and [64] gave theoretical evidence that training a robust model may lead to an increase of its standard risk. Hence finding a good approximation for Problem (3) may lead to a poor solution for Problem (1). Accordingly, it is natural to wonder whether we can ***find a class of models $\mathcal{H}$ for which we can control both the standard and adversarial risks?***

In this paper, we provide answers to the above question by conducting an in depth analysis of a special class of models called randomized classifiers, *i.e.* classifiers that output random variables instead of labels. Our main contributions summarize as follows.

1.3 Contributions

Our first contribution consists in studying randomized classifiers. By analogy with the deterministic case, we define a notion of robustness for randomized classifiers. This definition amounts to making the classifier locally Lipschitz with respect to the ℓ_p norm on $\mathcal{X}$, and a probability metric on $\mathcal{Y}$ (*e.g.* the total variation distance or the Renyi divergence). More precisely, if we denote D the probability metric at hand, a randomized classifier m is called (α_p, ϵ)-robust *w.r.t.* D if for any $\boldsymbol{x}, \boldsymbol{x}' \in \mathcal{X}$

$$\|\boldsymbol{x} - \boldsymbol{x}'\|_p \leq \alpha_p \implies D(\mathrm{m}(\boldsymbol{x}), \mathrm{m}(\boldsymbol{x}')) \leq \epsilon.$$

Denoting $\mathcal{M}_D(\alpha_p, \epsilon)$ the class of randomized classifiers that respect this local Lipschitz condition, we present the following results.

1. If D is either the total variation distance or the Renyi divergence, we show that for any $\mathrm{m} \in \mathcal{M}_D(\alpha_p, \epsilon)$, we can upper-bound the gap between the risk and the adversarial risk of m. Notably, if D is the total variation distance, for any $\mathrm{m} \in \mathcal{M}_D(\alpha_p, \epsilon)$ we have $\mathcal{R}^{\mathrm{adv}}(\mathrm{m}; \alpha_p) - \mathcal{R}(\mathrm{m}) \leq \epsilon$. Hence, ϵ controls the maximal trade-off between robust and standard accuracy for locally Lipschitz randomized classifier. We demonstrate similar results when D is the Renyi divergence showing that $\mathcal{R}^{\mathrm{adv}}(\mathrm{m}; \alpha_p) - \mathcal{R}(\mathrm{m}) \leq 1 - O\left(e^{-\epsilon}\right)$. This means that, for the class of locally Lipschitz randomized classifiers, solving the risk minimization problem, *i.e.* Problem (1), gives an approximate solution to the adversarial risk minimization problem, *i.e.* Problem (3), up to an additive factor that depends on the robustness parameter ϵ.

2. We devise an upper-bound on the generalization gap of any m in $\mathcal{M}_D(\alpha_p, \epsilon)$. In particular, when D is the total variation distance, we demonstrate that for any $\mathrm{m} \in \mathcal{M}_D(\alpha_p, \epsilon)$ we have

$$\mathcal{R}(\mathrm{m}) - \mathcal{R}_{\mathcal{S}}(\mathrm{m}) \leq O\left(\sqrt{\frac{N \times K}{n}}\right) + \epsilon,$$

   where N is the external α_p-covering number of the input samples. This means that, when $N/n \underset{n\to\infty}{\to} 0$, solving the empirical risk minimization problem, *i.e.* Problem (2), on $\mathcal{M}_D(\alpha_p, \epsilon)$ provides an approximate solution to the risk minimization problem, *i.e.* Problem (1). Since we can also bound the gap between the adversarial and the standard risk, we can combine the two results to bound the adversarial generalization gap on $\mathcal{M}_D(\alpha_p, \epsilon)$. Note however, that this result relies on a strong assumption on $\mathcal{X}$ that does not always avoid dimensionality issues. The problem of finding a subclass of $\mathcal{M}_D(\alpha_p, \epsilon)$ that provides tighter generalization bounds is an open question.

For our second contribution, we present a practical way to design this class $\mathcal{M}(\alpha_p, \epsilon)$ by using a simple yet efficient noise injection scheme. This allows us to build randomized classifiers from state-of-the-art machine learning models, including deep neural networks. More precisely our contribution is as follows.

1. Based on information-theoretic properties of the total variation distance and the Renyi divergence (*e.g.* the data processing inequality) we design a noise injection scheme to turn a state-of-the-art machine learning model into a robust randomized classifier. More formally, Let us denote Φ the c.d.f. of a standard Gaussian distribution. Let us consider $\boldsymbol{h}$ a deterministic hypothesis, we show that the randomized classifier $\mathrm{m} : \boldsymbol{x} \mapsto \boldsymbol{h}(\boldsymbol{x} + n)$ with $n \sim \mathcal{N}(0, \sigma^2 I_d)$ is both $(\alpha_2, \frac{(\alpha_2)^2}{2\sigma})$-robust *w.r.t.* the Renyi divergence and $(\alpha_2,\ 2\Phi\left(\frac{\alpha_2}{2\sigma}\right) - 1)$-robust *w.r.t.* the total variation distance. Our results on randomized classifiers are applicable to a wide range of machine learning models including deep neural networks.

2. We further corroborate our theoretical results with experiments using deep neural networks on standard image datasets, namely CIFAR-10 and CIFAR-100 [26]. These models can simultaneously provide accurate prediction (over 0.82 clean accuracy on CIFAR-10) and reasonable robustness against ℓ_2 adversarial examples (0.45 against ℓ_2 adversaries with magnitude 0.5 on CIFAR-10).

2 Related Work

Contrary to other notions such as training corruption, *a.k.a.* poisoning attacks [23, 24], the theoretical study of adversarial robustness is still in its infancy. So far, empirical observations tend to show that 1) adversarial examples on state-of-the-art models are hard to mitigate and 2) robust training methods give poor generalization performances. Some recent works started to study the problem through the lens of learning theory either to understand the links between robustness and accuracy or to provide bounds on the generalization gap of current learning procedures in the adversarial setting.

2.1 Accuracy vs robustness trade-off

A first line of research [22, 49, 52] suggests that designing robust models might be inconsistent with standard accuracy. These works argue with experiments and toy examples that robust and standard classification are two concurrent problems. Following this line, [64] observed that the adversarial risk of any hypothesis $\boldsymbol{h}$ decomposes as follows,

$$\mathcal{R}^{\mathrm{adv}}(\boldsymbol{h}; \alpha_p) = \mathcal{R}(\boldsymbol{h}) + \mathcal{R}^{\mathrm{adv}}_{>0}(\boldsymbol{h}; \alpha_p), \tag{5}$$

where $\mathcal{R}^{\mathrm{adv}}_{>0}(\mathrm{m}; \alpha_p)$ is the amount of risk that the adversary gets with *non-null* perturbations. Looking at Equation (5), we realize that minimizing the adversarial risk is not enough to control standard accuracy, as one could only optimize over the second term. This indicates that adversarial risk minimization, *i.e.* Problem (3), is harder to solve than the standard risk minimization, *i.e.* Problem (1).

While this indicates that both goals maybe difficult be achieve simultaneously, Equation (5), along with the empirical studies from the literature do not highlight any fundamental trade-off between robustness and accuracy. Moreover, no upper-bound on $\mathcal{R}^{\mathrm{adv}}_{>0}(\boldsymbol{h}; \alpha_p)$ has been demonstrated yet. Hence the questions whether this trade-off exists and can be controlled remain open. In this paper, we provide a rigorous answer to these questions by identifying classes $\mathcal{M}_D(\alpha_p, \epsilon)$ of randomized classifiers for which we can upper bound the trade-off term $\mathcal{R}^{\mathrm{adv}}_{>0}(\mathrm{m}; \alpha_p)$ for any $\mathrm{m} \in \mathcal{M}_D(\alpha_p, \epsilon)$. This shows that for some classes of randomized classifiers, precision is not conflicting with robustness, since we can control the maximum loss of accuracy that the model can suffer in the adversarial setting. It also challenges the intuitions developed by previous works [22, 49, 52] and argues in favor of using randomized mechanisms as a defense against adversarial attacks.

2.2 Studying adversarial generalization

To further compare the hardness of the two problems, a recent line of research began to explore the notion of adversarial generalization gap. In this line, [44] presented some first intuitions by studying a simplified binary classification framework where $\mathcal{D}$ is a mixture of multi-dimensional Gaussian distributions. In this framework the authors show that without attacks, we only need $O(1)$ training samples to have a small generalization gap. But against an ℓ_∞ adversary, we need $O(\sqrt{d})$ training samples instead. In the discussion

of their work, the authors present the problem of obtaining similar results without making any assumption about the distribution as an open problem.

This issue was recently studied using the Rademacher complexity by [25, 62] and [2]. These papers relate the adversarial generalization error of linear classifiers and one-hidden layer neural networks with the dimension of the problem. They show that the adversarial generalization depends on the dimension of the problem. At a first glance, the difficulty of adversarial generalization seems to contradict previous conclusions on the link between robustness and generalization presented by [60]. But, as we will discuss in the sequel, these results assume that the input space $\mathcal{X}$ can be partitioned in $O(1)$ sub-space in which the classification function has small variations. This assumption may not always hold when dealing with high dimensional input spaces (*e.g.* images) and very sophisticated classification algorithms (*e.g.* deep neural networks).

Going further, it should be noted that the generalization gap measures only the difference between empirical and theoretical risks. In practice, the empirical adversarial risk is hard to estimate, since we cannot compute the exact solution to the inner maximization problem. The following question therefore remains open: even if we can set up a learning procedure with a controlled generalization gap, can we give guarantees on the standard and adversarial risks? In this paper, we start answering this question by providing techniques that provably offer both small standard risk and reasonable robustness against adversarial examples (see Section 1.3 for more details).

2.3 Defense against adversarial examples based on noise injection

Injecting noise into algorithms to improve train time robustness has been used for ages in detection and signal processing tasks [7, 19, 34, 65]. It has also been extensively studied in several machine learning and optimization fields, *e.g.* robust optimization [4] and data augmentation techniques [37]. Concurrently to our work, noise injection techniques have been adopted by the adversarial defense community under the *randomized smoothing* name. The idea of provable defense through noise injection was first proposed by [28] and refined by [9, 29] and [43]. The rational behind randomized smoothing is very simple: smooth $\boldsymbol{h}$ *after training* by convolution with a Gaussian measure to build a more stable classifier. Our work belongs to the same line of research, but the nature of our results is different. While randomized smoothing focuses on the construction of certified defenses, depending on the dataset and the classifier at hand, we study the generalization properties of randomized mechanisms both in the standard and the adversarial setting. Our analysis presents the fundamental properties of randomized defenses, including (but not limited to) randomized smoothing (c.f. Section 7).

3 Definition of Risk and Robustness for Randomized classifiers

In this work, the goal is to analyze how randomized classifiers can solve the problem of classification in the presence of an adversary. Let us start by defining what we mean by randomized classifiers.

Remark 1 (Remark on measurability) *Through the paper, we assume every spaces $\mathcal{Z}$ to be associated with a σ-algebra denoted $\mathcal{A}(\mathcal{Z})$. Furthermore, we denote $\mathcal{P}(\mathcal{Z})$ the set of probability distributions defined on the measurable space $(\mathcal{Z}, \mathcal{A}(\mathcal{Z}))$. In the following, for simplicity, we refer to $\mathcal{A}(\mathcal{Z})$ only when necessary.*

Definition 1 (Probabilistic mapping) *Let $\mathcal{Z}$ and $\mathcal{Z}'$ be two arbitrary spaces. A* probabilistic mapping *from $\mathcal{Z}$ to $\mathcal{Z}'$ is a mapping $\mathrm{m} : \mathcal{Z} \to \mathcal{P}(\mathcal{Z}')$, where $\mathcal{P}(\mathcal{Z}')$ is the space of probability measures on $\mathcal{Z}'$. When $\mathcal{Z} = \mathcal{X}$ and $\mathcal{Z}' = \mathcal{Y}$, m is called a* randomized classifier*. To get a numerical answer out of m for an input $\boldsymbol{x}$, we sample $\hat{y} \sim \mathrm{m}(\boldsymbol{x})$.*

Any mapping can be considered as a probabilistic mapping, whether it explicitly considers randomization or not. In fact, any deterministic classifier can be considered as a randomized one, since it can be characterized by a Dirac measure. Accordingly, the definition of a randomized classifier is fully general and equally consider classifiers with or without randomization scheme.

3.1 Risk and adversarial risk for randomized classifiers

To analyze this new hypothesis class, we can adapt the concepts of risk and adversarial risk for a randomized classifier. The loss function we use is the natural extension of the 0/1 loss to the randomized regime. Given a randomized classifier m and a sample $(\boldsymbol{x}, y) \sim \mathcal{D}$ it writes

$$\mathcal{L}_{0/1}(\mathrm{m}(\boldsymbol{x}), y) := \mathbb{E}_{\hat{y}\sim\mathrm{m}(\boldsymbol{x})}\left[\mathbb{1}\left\{\hat{y} \neq y\right\}\right]. \tag{6}$$

This loss function evaluates the probability of misclassification of m on a data sample $(\boldsymbol{x}, y) \sim \mathcal{D}$. Accordingly, the risk of m with respect to $\mathcal{D}$ writes

$$\mathcal{R}(\mathrm{m}) := \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[\mathcal{L}_{0/1}(\mathrm{m}(\boldsymbol{x}), y)\right]. \tag{7}$$

Finally, given m and $(\boldsymbol{x}, y) \sim \mathcal{D}$, the adversary seeks a perturbation $\boldsymbol{\tau} \in B_p(\alpha_p)$ that maximizes the expected error of the classifier on $\boldsymbol{x}$ (*i.e.* $\mathbb{E}_{\hat{y}\sim\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau})}\left[\mathbb{1}\left\{\hat{y} \neq y\right\}\right]$). Therefore, the adversarial risk of m under α_p-bounded perturbations writes

$$\mathcal{R}^{\mathrm{adv}}(\mathrm{m}; \alpha_p) := \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[\sup_{\boldsymbol{\tau}\in B_p(\alpha_p)} \mathcal{L}_{0/1}(\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau}), y)\right]. \tag{8}$$

By analogy with the deterministic setting, we denote $\mathcal{R}_{\mathcal{S}}(\mathrm{m}) := \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}_{0/1}(\mathrm{m}(\boldsymbol{x}_i), y_i)$ and $\mathcal{R}^{\mathrm{adv}}_{\mathcal{S}}(\mathrm{m}; \alpha_p) := \frac{1}{n}\sum_{i=1}^{n} \sup_{\boldsymbol{\tau}\in B_p(\alpha_p)} \mathcal{L}_{0/1}(\mathrm{m}(\boldsymbol{x_i}+\boldsymbol{\tau}), y_i)$ the empirical risks of m for a given training sample $\mathcal{S} := \{(\boldsymbol{x_1}, y_1), \ldots, (\boldsymbol{x_n}, y_n)\}$.

3.2 Robustness for randomized classifiers

We could define the notion of robustness for a randomized classifier depending on whether it misclassifies any test sample $(\boldsymbol{x}, y) \sim \mathcal{D}$. But in practice, neither the adversary nor the model provider have access to the ground-truth distribution $\mathcal{D}$. Furthermore, in real-world scenarios, one wants to check before its deployment that the model is robust. Therefore, it is required for the classifier to be stable on the regions of the space where it already classifies correctly. Formally a (deterministic) classifier $c : \mathcal{X} \to \mathcal{Y}$ is called *robust* if for any $(\boldsymbol{x}, y) \sim \mathcal{D}$ such that $c(\boldsymbol{x}) = y$, and for any $\boldsymbol{\tau} \in \mathcal{X}$ one has

$$\|\boldsymbol{\tau}\|_p \leq \alpha_p \implies c(\boldsymbol{x}) = c(\boldsymbol{x}+\boldsymbol{\tau}). \tag{9}$$

By analogy with this notion, we define robustness for a randomized classifier as follows.

Definition 2 (Robustness for a randomized classifier) *A randomized classifier* $\mathrm{m} : \mathcal{X} \to \mathcal{P}(\mathcal{Y})$ *is called* (α_p, ϵ)-robust *w.r.t. D if for any* $\boldsymbol{x}, \boldsymbol{\tau} \in \mathcal{X}$*, one has*

$$\|\boldsymbol{\tau}\|_p \leq \alpha_p \implies D\left(\mathrm{m}(\boldsymbol{x}), \mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau})\right) \leq \epsilon\,.$$

Where D is a metric/divergence between two probability measures. Given such a metric/divergence D, we denote $\mathcal{M}_D(\alpha_p, \epsilon)$ *the set of all randomized classifiers that are* (α_p, ϵ)-robust *w.r.t. D.*

Note that we did not add the constraint that m classifies well on $(\boldsymbol{x}, y) \sim \mathcal{D}$, since it is already encompassed in the probability distribution itself. If the two probabilities $\mathrm{m}(\boldsymbol{x})$ and $\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau})$ are close, and if $\mathrm{m}(\boldsymbol{x})$ outputs y with high probability, then it will be the same for $\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau})$. This formulation naturally raises the question of the choice of the metric D. Any choice of metric/divergence will instantiate a notion of adversarial robustness, and it should be carefully selected. In the present work, we focus our study on the total variation distance and the Renyi divergence. The question whether these metrics/divergences are more appropriate than others remains open but these two divergences are sufficiently general to cover a wide range of other definitions (see Appendix A for more details). Furthermore, these notions of distance comply with both a theoretical analysis (Section 5) and practical considerations (Section 8).

3.3 Divergence and metrics between probability measures.

Let us now recall the definition of total variation distance and Renyi divergence. Let $\mathcal{Z}$ be an arbitrary space, and ρ, ρ' be two measures in $\mathcal{P}(\mathcal{Z})$[1]. The *total variation distance* between ρ and ρ' is

$$D_{TV}\left(\rho,\rho'\right) := \sup_{Z\subset\mathcal{A}(\mathcal{Z})} \left|\rho(Z)-\rho'(Z)\right| \ , \tag{10}$$

where $\mathcal{A}(\mathcal{Z})$ is the σ-algebra associated with the set of measures $\mathcal{P}(\mathcal{Z})$. The total variation distance is one of the most commonly used probability metrics. It admits several very simple interpretations, and is a very useful tool in many mathematical fields such as probability theory, Bayesian statistics or optimal transport [38, 42, 57]. In optimal transport, it can be rewritten as the solution of the Monge-Kantorovich problem with the cost function $\text{cost}(\boldsymbol{z},\boldsymbol{z}') = \mathbb{1}\left\{\boldsymbol{z}\neq\boldsymbol{z}'\right\}$,

$$D_{TV}(\rho,\rho') = \inf \int_{\mathcal{Z}^2} \mathbb{1}\left\{\boldsymbol{z}\neq\boldsymbol{z}'\right\} d\pi(\boldsymbol{z},\boldsymbol{z}') \ , \tag{11}$$

where the infimum is taken over all joint probability measures π in $\mathcal{P}\left(\mathcal{Z}\times\mathcal{Z}\right)$ with marginals ρ and ρ'. According to this interpretation, it seems quite natural to consider the total variation distance as a relaxation of the trivial distance on $[0,1]$ (for deterministic classifiers).

Let us now suppose that ρ and ρ' admit probability density functions g and g' according to a third measure ν. Then the *Renyi divergence of order* β between ρ and ρ' writes

$$D_{\beta}\left(\rho,\rho'\right) := \frac{1}{\beta-1}\log\int_{\mathcal{Y}} g'(y)\left(\frac{g(y)}{g'(y)}\right)^{\beta} d\nu(y) \ . \tag{12}$$

The Renyi divergence [41] is a generalized divergence defined for any β on the interval $[1,\infty]$. It equals the Kullback-Leibler divergence when $\beta\to 1$, and the maximum divergence when $\beta\to\infty$. It also has the property of being non-decreasing with respect to β. This divergence is very common in machine learning and Information theory [55], especially in its Kullback-Leibler form as it is widely used as the loss function, *i.e.* cross entropy, of classification algorithms. In the remaining, we denote $\mathcal{M}_{\beta}\left(\alpha_p,\epsilon\right)$ the set of (α_p,ϵ)-robust classifiers w.r.t. D_{β}.

Let us now give some properties of these divergences that will be useful for our analysis. First we recall the probability preservation property of the Renyi divergence, first presented by [27].

Proposition 1 ([27]) *Let ρ and ρ' be two measures in $\mathcal{P}(\mathcal{Z})$. Then for any $Z\in\mathcal{A}(\mathcal{Z})$, the following holds,*

$$\rho(Z) \leq \left(\exp\left(D_{\beta}(\rho,\rho')\right)\rho'(Z)\right)^{\frac{\beta-1}{\beta}} .$$

Now thanks to previous works by [16] and [53], we also get the following results relating the total variation distance and the Renyi divergence.

Proposition 2 (Inequality between total variation and Renyi divergence) *Let ρ and ρ' be two measures in $\mathcal{P}(\mathcal{Z})$, and $\beta\geq 1$. Then the following holds,*

$$D_{TV}(\rho,\rho') \leq \min\left(\frac{3}{2}\left(\sqrt{1+\frac{4D_{\beta}(\rho,\rho')}{9}}-1\right)^{1/2}, \ \frac{\exp\left(D_{\beta}(\rho,\rho')+1\right)-1}{\exp\left(D_{\beta}(\rho,\rho')+1\right)+1}\right).$$

Proof 1 *Thanks to [16], one has*

$$D_1(\rho,\rho') \geq 2D_{TV}(\rho,\rho')^2 + \frac{4D_{TV}(\rho,\rho')^4}{9}.$$

[1]Recall from Definition 1 that $\mathcal{P}(\mathcal{Z})$ is the set of probability measures on $\mathcal{Z}$

From which it follows that

$$D_{TV}(\rho,\rho') \leq \frac{3}{2}\left(\sqrt{1+\frac{4D_1(\rho,\rho')}{9}}-1\right)^{1/2}.$$

Moreover, using inequality from [53], one gets

$$D_1(\rho,\rho')+1 \geq \log\left(\frac{1+D_{TV}(\rho,\rho')}{1-D_{TV}(\rho,\rho')}\right).$$

This inequality leads to the following

$$\frac{\exp(D_1(\rho,\rho')+1)-1}{\exp(D_1(\rho,\rho')+1)+1} \geq D_{TV}(\rho,\rho').$$

By combining the above inequalities and by monotony of Renyi divergence regarding β, one obtains the expected result.

From now on, we denote $\mathcal{M}_{TV}(\alpha,\epsilon)$ and $\mathcal{M}_{\beta}(\alpha,\epsilon)$ the set of (α,ϵ)-*robust* classifiers respectively for D_{TV} and D_{β}. The next section gives bounds on the generalization gap in the standard and the adversarial settings for these specific hypothesis classes.

4 Risks' gap and Generalization gap for randomized classifiers

As discussed in Section 2.1, we can always decompose the adversarial risk of a classifier $\mathcal{R}^{\text{adv}}(\mathrm{m};\alpha_p)$ in two terms. First the standard risk $\mathcal{R}(\mathrm{m})$ and second the amount of risk the adversary creates with non-zero perturbations $\mathcal{R}^{\text{adv}}_{>0}(\mathrm{m};\alpha_p)$. Hence minimizing $\mathcal{R}(\mathrm{m})$ can give poor values for $\mathcal{R}^{\text{adv}}(\mathrm{m};\alpha_p)$ and vice-versa. In this section, we upper-bound the risks' gap $\mathcal{R}^{\text{adv}}_{>0}(\mathrm{m};\alpha_p)$, *i.e.* the gap between the risk and the adversarial risk of a robust classifier.

4.1 Risks' gap for robust classifiers w.r.t. D_{TV}

First, let us consider $\mathrm{m} \in \mathcal{M}_{TV}(\alpha_p,\epsilon)$. We can control the loss of accuracy under attack of this classifier with the robustness parameter ϵ.

Theorem 1 (Risk's gap for robust classifiers w.r.t D_{TV}) *Let $\mathrm{m} \in \mathcal{M}_{TV}(\alpha_p,\epsilon)$. Then we have*

$$\mathcal{R}^{\text{adv}}(\mathrm{m};\alpha_p) \leq \mathcal{R}(\mathrm{m}) + \epsilon\,.$$

Proof 2 *Let* m *be an (α_p,ϵ)-robust classifier* w.r.t. D_{TV} *, $(\boldsymbol{x},y)\sim\mathcal{D}$ and $\boldsymbol{\tau}\in\mathcal{X}$ such that $\|\boldsymbol{\tau}\|_p \leq \alpha_p$. By definition of the $0/1$ loss we have*

$$\mathcal{L}_{0/1}(\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau}),y) = \mathbb{E}_{\hat{y}\sim\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau})}\left[\mathbb{1}\left\{\hat{y}\neq y\right\}\right].$$

Furthermore, by definition of the total variation distance we have

$$\mathbb{E}_{\hat{y}\sim\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau})}\left[\mathbb{1}\left\{\hat{y}\neq y\right\}\right] - \mathbb{E}_{\hat{y}\sim\mathrm{m}(\boldsymbol{x})}\left[\mathbb{1}\left\{\hat{y}\neq y\right\}\right] \leq D_{TV}(\mathrm{m}(\boldsymbol{x}),\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau})).$$

Since $\mathrm{m} \in \mathcal{M}_{TV}(\alpha_p,\epsilon)$, the above amounts to write

$$\mathcal{L}_{0/1}(\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau}),y) - \mathcal{L}_{0/1}(\mathrm{m}(\boldsymbol{x}),y) \leq \epsilon.$$

Finally, this holds for any $(\boldsymbol{x},y)\sim\mathcal{D}$ and any α_p bounded perturbation $\boldsymbol{\tau}$, then we get

$$\mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[\sup_{\boldsymbol{\tau}\in B_p(\alpha_p)} \mathcal{L}_{0/1}(\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau}),y)\right] - \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[\mathcal{L}_{0/1}(\mathrm{m}(\boldsymbol{x}),y)\right] \leq \epsilon.$$

The above inequality concludes the proof.

This result means that if we can design a class $\mathcal{M}_{TV}(\alpha_p, \epsilon)$ with small enough ϵ, then minimizing the risk of $\mathrm{m} \in \mathcal{M}_{TV}(\alpha_p, \epsilon)$ is also sufficient to control the adversarial risk. It is relatively easy to obtain, but it has an interesting consequence on the understanding we have of the trade-off between robustness and accuracy. It says that there exists some classes of randomized classifiers for which robustness and standard accuracy may not be at odds, since we can upper-bound the maximal loss of accuracy the model may suffer under attack. This questions previous intuitions developed on deterministic classifiers by [22, 49, 52] and [64] and advocates for the use of randomization schemes as defenses against adversarial attacks. Note, however, that we did not evade the trade-off between robustness and accuracy, we only showed that with certain hypothesis classes it can be controlled.

4.2 Risks' gap for robust classifiers w.r.t. D_β

We now extend the previous results the Renyi divergence. We show that, for any randomized classifier in $\mathcal{M}_\beta(\alpha_p, \epsilon)$, we can bound the gap between the risk and the adversarial risk of m. Using the Renyi divergence, the factor that controls the classifier's loss of accuracy under attack can be either multiplicative or additive, and depends both on the robustness parameter ϵ and on the divergence parameter β.

Theorem 2 (Multiplicative risks' gap for Renyi-robust classifiers) *Let* $\mathrm{m} \in \mathcal{M}_\beta(\alpha_p, \epsilon)$. *Then we have*

$$\mathcal{R}^{\mathrm{adv}}(\mathrm{m}; \alpha_p) \leq \left(e^\epsilon \mathcal{R}(\mathrm{m})\right)^{\frac{\beta-1}{\beta}} .$$

Proof 3 *Let* m *be an* (α_p, ϵ)*-robust classifier* w.r.t. D_β, $(\boldsymbol{x}, y) \sim \mathcal{D}$ *and* $\boldsymbol{\tau} \in \mathcal{X}$ *such that* $\|\boldsymbol{\tau}\|_p \leq \alpha_p$. *With the same reasoning as above, and with Proposition 1, we get*

$$\begin{aligned}
\mathcal{L}_{0/1}\left(\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau}), y\right) &= \mathbb{E}_{\hat{y}\sim\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau})}\left[\mathbb{1}\left\{\hat{y} \neq y\right\}\right] \\
&= \mathbb{P}_{\hat{y}\sim\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau})}\left[\hat{y} \neq y\right] \\
&\leq \left(e^{D_\beta(\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau}),\mathrm{m}(\boldsymbol{x}))}\mathbb{P}_{\hat{y}\sim\mathrm{m}(\boldsymbol{x})}\left[\hat{y} \neq y\right]\right)^{\frac{\beta-1}{\beta}} \quad \textit{(Prop. 1)} \\
&= \left(e^{D_\beta(\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau}),\mathrm{m}(\boldsymbol{x}))}\mathbb{E}_{\hat{y}\sim\mathrm{m}(\boldsymbol{x})}\left[\mathbb{1}\left\{\hat{y} \neq y\right\}\right]\right)^{\frac{\beta-1}{\beta}} \\
&\leq \left(e^\epsilon \mathcal{L}_{0/1}\left(\mathrm{m}(\boldsymbol{x}), y\right)\right)^{\frac{\beta-1}{\beta}} .
\end{aligned}$$

Since this holds for any $(\boldsymbol{x}, y) \sim \mathcal{D}$ *and any* α_p *bounded perturbation* $\boldsymbol{\tau}$, *we get*

$$\begin{aligned}
\mathcal{R}^{\mathrm{adv}}(\mathrm{m}; \alpha_p) &= \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[\sup_{\boldsymbol{\tau}\in B_p(\alpha_p)} \mathcal{L}_{0/1}\left(\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau}), y\right)\right] \\
&\leq \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[e^{\frac{\beta-1}{\beta}\epsilon}\mathcal{L}_{0/1}\left(\mathrm{m}(\boldsymbol{x}), y\right)^{\frac{\beta-1}{\beta}}\right] \\
&\leq e^{\frac{\beta-1}{\beta}\epsilon}\mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[\mathcal{L}_{0/1}\left(\mathrm{m}(\boldsymbol{x}), y\right)^{\frac{\beta-1}{\beta}}\right] .
\end{aligned}$$

Finally, using the Jensen inequality, one gets

$$\leq e^{\frac{\beta-1}{\beta}\epsilon}\mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[\mathcal{L}_{0/1}\left(\mathrm{m}(\boldsymbol{x}), y\right)\right]^{\frac{\beta-1}{\beta}} = \left(e^\epsilon \mathcal{R}(\mathrm{m})\right)^{\frac{\beta-1}{\beta}} .$$

The above inequality concludes the proof.

This first result gives a multiplicative bound on the gap between the standard and adversarial risks. This means that if we can design a class $\mathcal{M}_\beta(\alpha_p, \epsilon)$ with small enough ϵ, and big enough β, then minimizing the risk of any $\mathrm{m} \in \mathcal{M}_\beta(\alpha_p, \epsilon)$ is sufficient to also minimize the adversarial risk of m. Nevertheless, multiplicative factors are not easy to analyze.

Remark 2 *More general bounds can be computed if we assume that for every randomized classifier* m *there exists a convex function* $\mathbf{f}$ *such that for all* $\boldsymbol{x}$ *and* $\boldsymbol{\tau}$ *with* $\|\boldsymbol{\tau}\|_p \leq \alpha_p$*, we have* $\mathrm{m}(\boldsymbol{x})(Z) \leq \mathbf{f}(\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau})(Z))$ *for all measurable sets* Z*. In this case, we get* $\mathcal{R}^{\mathrm{adv}}(\mathrm{m};\alpha_p) \leq \mathbf{f}\left(\mathcal{R}(\mathrm{m})\right)$*. This has a close link with randomized smoothing [9] and* f*-differential privacy [14] where both try to fit the best possible* $\mathbf{f}$ *using Neyman-Pearson lemma.*

The following result provides an additive counterpart to Theorem 2. It gives a control over the loss of accuracy under attack with respect to the robustness parameter ϵ and the Shannon entropy of m.

Theorem 3 (Additive risks' gap for Renyi-robust classifiers) *Let* $\mathrm{m} \in \mathcal{M}_\beta\left(\alpha_p, \epsilon\right)$*, then we have*

$$\mathcal{R}^{\mathrm{adv}}(\mathrm{m};\alpha_p) - \mathcal{R}(\mathrm{m}) \leq 1 - e^{-\epsilon}\mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_{|\mathcal{X}}}\left[e^{-H(\mathrm{m}(\boldsymbol{x}))}\right]$$

where H *is the Shannon entropy (*i.e. *for any* $\rho \in \mathcal{P}\left(\mathcal{Y}\right), H(\rho) = -\sum_{k\in\mathcal{Y}} \rho_k \log(\rho_k)$*) and* $\mathcal{D}_{|\mathcal{X}}$ *is the marginal distribution of* $\mathcal{D}$ *for* $\mathcal{X}$*.*

Proof 4 *Let* $\mathrm{m} \in \mathcal{M}_\beta\left(\alpha_p, \epsilon\right)$*, then*

$$\begin{aligned}
& \mathcal{R}^{\mathrm{adv}}(\mathrm{m};\alpha_p) - \mathcal{R}(\mathrm{m}) \\
= \ & \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[\sup_{\boldsymbol{\tau}\in B_p(\alpha_p)} \mathcal{L}_{0/1}\left(\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau}), y\right) - \mathcal{L}_{0/1}\left(\mathrm{m}(\boldsymbol{x}), y\right)\right].
\end{aligned}$$

By definition of the $0/1$ *loss, this amounts to write*

$$\begin{aligned}
= \ & \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[\sup_{\boldsymbol{\tau}\in B_p(\alpha_p)} \mathbb{E}_{\hat{y}_{adv}\sim\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau}),\hat{y}\sim\mathrm{m}(\boldsymbol{x})}\left[\mathbb{1}\left(\hat{y}_{adv} \neq y\right) - \mathbb{1}\left(\hat{y} \neq y\right)\right]\right] \\
\leq \ & \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[\sup_{\boldsymbol{\tau}\in B_p(\alpha_p)} \mathbb{E}_{\hat{y}_{adv}\sim\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau}),\hat{y}\sim\mathrm{m}(\boldsymbol{x})}\left[\mathbb{1}\left(\hat{y}_{adv} \neq \hat{y}\right)\right]\right] \\
= \ & \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[\sup_{\boldsymbol{\tau}\in B_p(\alpha_p)} \mathbb{P}_{\hat{y}_{adv}\sim\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau}),\hat{y}\sim\mathrm{m}(\boldsymbol{x})}\left[\hat{y}_{adv} \neq \hat{y}\right]\right] \\
= \ & \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[\sup_{\boldsymbol{\tau}\in B_p(\alpha_p)} 1 - \mathbb{P}_{\hat{y}_{adv}\sim\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau}),\hat{y}\sim\mathrm{m}(\boldsymbol{x})}\left[\hat{y}_{adv} = \hat{y}\right]\right] \\
= \ & \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[\sup_{\boldsymbol{\tau}\in B_p(\alpha_p)} 1 - \sum_{i=1}^{K} \mathrm{m}(\boldsymbol{x})_i \times \mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau})_i\right].
\end{aligned}$$

Now, note that for any $(\boldsymbol{x}, y) \sim \mathcal{D}$ *and* $\boldsymbol{\tau} \in \mathcal{X}$*, by definition of a probability vector in* $\mathcal{P}\left(\mathcal{Y}\right)$*, and thanks to Jensen inequality we can write*

$$\sum_{i=1}^{K} \mathrm{m}(\boldsymbol{x})_i \times \mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau})_i \geq \exp\left(\sum_{i=1}^{K} \mathrm{m}(\boldsymbol{x})_i \log \mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau})_i\right).$$

Then by definition of the entropy and the Kullback Leibler divergence we have

$$\exp\left(\sum_{i=1}^{K} \mathrm{m}(\boldsymbol{x})_i \log \mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau})_i\right) = \exp\big(-D_1\left(\mathrm{m}(\boldsymbol{x}), \mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau})\right) - H\left(\mathrm{m}(\boldsymbol{x})\right)\big).$$

Finally, by combining the above inequalities and since $\mathrm{m} \in \mathcal{M}_\beta\left(\alpha_p, \epsilon\right)$ *we get*

$$\mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[\sup_{\boldsymbol{\tau}\in B_p(\alpha_p)} \mathbb{P}_{\hat{y}_{adv}\sim\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau}),\hat{y}\sim\mathrm{m}(\boldsymbol{x})}(\hat{y}_{adv} \neq \hat{y})\right]$$

$$\leq \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[\sup_{\boldsymbol{\tau}\in B_p(\alpha_p)} 1 - e^{-D_1(\mathrm{m}(\boldsymbol{x}),\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau}))-H(\mathrm{m}(\boldsymbol{x}))}\right]$$
$$\leq \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[1 - e^{-\epsilon-H(\mathrm{m}(\boldsymbol{x}))}\right] = 1 - e^{-\epsilon}\mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_{|\mathcal{X}}}\left[e^{-H(\mathrm{m}(\boldsymbol{x}))}\right].$$

The above inequality concludes the proof.

This result is interesting because it relates the accuracy of m with the bound we obtain. In words, when $\mathrm{m}(\boldsymbol{x})$ has large entropy (*i.e.* $H(\mathrm{m}(\boldsymbol{x})) \to \log(K)$) the output distribution tends towards the uniform distribution; hence $\epsilon \to 0$. This means that the classifier is very robust but also completely inaccurate, since it outputs classes uniformly at random. On the opposite, if $H(\mathrm{m}(\boldsymbol{x})) \to 0$, then $\epsilon \to \infty$. The classifier may be accurate, but it is not robust anymore (at least according to our definition). Hence we need to find a classifier that achieves a trade-off between robustness and accuracy.

5 Standard Generalization gap

In this section we devise generalization gap bounds for randomized classifiers when they are robust according either to the total variation distance or the Renyi divergence. To do so, we upper-bound the Rademacher complexity of the loss space for TV-robust classifiers

$$\mathcal{L}_{\mathcal{M}_{TV}(\alpha_p,\epsilon)} := \{(\boldsymbol{x},y) \mapsto \mathcal{L}_{0/1}(\boldsymbol{h}(\boldsymbol{x}),y) \mid \mathrm{m} \in \mathcal{M}_{TV}(\alpha_p,\epsilon)\}.$$

The *empirical Rademacher complexity*, first introduced by [3], is one of the standard measures of generalization gap. It is particularly useful to obtain quality bounds for complex classes such as neural networks since it does not depend on the number of parameters in the network contrary to combinatorial notions such as the *VC dimension*.

Definition 3 (Rademacher complexity) *For any class of real-valued functions $\mathcal{F} := \{(\boldsymbol{x},y) \mapsto \mathbb{R}\}$, given a training sample $\mathcal{S} = \{(\boldsymbol{x_1},y_1),\ldots,(\boldsymbol{x_n},y_n)\}$, the* empirical Rademacher complexity *of $\mathcal{F}$ is defined as*

$$\mathfrak{R}_{\mathcal{S}}(\mathcal{F}) := \frac{1}{n}\mathbb{E}_{r_i}\left[\sup_{f\in\mathcal{F}}\sum_{i=1}^{n} r_i f(\boldsymbol{x_i},y_i)\right],$$

where r_i are i.i.d. *drawn from a Rademacher measure (*i.e. $\mathbb{P}(r_i=1) = \mathbb{P}(r_i=-1) = \frac{1}{2}$*).*

The empirical Rademacher complexity measures the uniform convergence rate of the empirical risk towards the risk on the function class $\mathcal{F}$ as demonstrated by [35]. Thanks to this notion of complexity, we can bound with high probability the generalization gap of any hypothesis m in a class $\mathcal{M}$.

Theorem 4 ([35]) *Let $\mathcal{M}$ be a class of possibly randomized classifiers and $\mathcal{L}_{\mathcal{M}} := \{\mathcal{L}_{\mathrm{m}} : (\boldsymbol{x},y) \mapsto \mathcal{L}_{0/1}(\mathrm{m}(\boldsymbol{x}),y) \mid \mathrm{m} \in \mathcal{M}\}$. Then for any $\delta \in (0,1)$, with probability at least $1-\delta$, the following holds for any $\mathrm{m} \in \mathcal{M}_{TV}(\alpha_p,\epsilon)$,*

$$\mathcal{R}(\mathrm{m}) - \mathcal{R}_{\mathcal{S}}(\mathrm{m}) \leq 2\mathfrak{R}_{\mathcal{S}}(\mathcal{L}_{\mathcal{M}}) + 3\sqrt{\frac{\ln(2/\delta)}{2n}}.$$

5.1 Generalization error for robust classifiers

Accordingly, we want to upper bound the empirical Rademacher complexity of $\mathcal{L}_{\mathcal{M}_{TV}(\alpha_p,\epsilon)}$, which motivates the following definition.

Definition 4 (α-covering and external covering number) *Let us consider $(\mathcal{X}, \|.\|_p)$ a vector space equipped with the ℓ_p norm, $B \subset \mathcal{X}$ and $\alpha \geq 0$. Then*

- *$C = \{\boldsymbol{c_1}, \dots, \boldsymbol{c_m}\}$ is an α-covering of B for the ℓ_p norm if for any $\boldsymbol{x} \in B$ there exists $\boldsymbol{c_i} \in C$ such that $\|\boldsymbol{x} - \boldsymbol{c_i}\|_p \leq \alpha$.*
- *The external covering number of B writes $N\left(B, \|.\|_p, \alpha\right)$. It is the minimal number of points one needs to build an α-covering of B for the ℓ_p norm.*

The covering number is a well-known measure that is often used in statistical learning theory [45] and asymptotic statistics [54] to evaluate the complexity of a set of functions. Here we use it to evaluate the number of ℓ_p balls we need to cover the training samples, which gives us the following bound on the Rademacher complexity of $\mathcal{L}_{\mathcal{M}_{TV}(\alpha_p,\epsilon)}$.

Theorem 5 (Rademacher complexity for TV-robust classifiers) *Let $\mathcal{L}_{\mathcal{M}_{TV}(\alpha_p,\epsilon)}$ be the loss function class associated with $\mathcal{M}_{TV}(\alpha_p, \epsilon)$. Then, for any $\mathcal{S} := \{(\boldsymbol{x_1}, y_1), \dots, (\boldsymbol{x_n}, y_n)\}$, the following holds,*

$$\mathfrak{R}_{\mathcal{S}}\left(\mathcal{L}_{\mathcal{M}_{TV}(\alpha_p,\epsilon)}\right) \leq \sqrt{\frac{N \times K}{n}} + \epsilon.$$

Where $N = N\left(\{\boldsymbol{x_1}, \dots, \boldsymbol{x_n}\}, \|.\|_p, \alpha_p\right)$ is the α_p-external covering number of the inputs $\{\boldsymbol{x_1}, \dots, \boldsymbol{x_n}\}$ for the ℓ_p norm.

Proof 5 *Let us denote $\mathcal{S} := \{(\boldsymbol{x_1}, y_1), \dots, (\boldsymbol{x_n}, y_n)\}$ and $N = N\left(\{\boldsymbol{x_1}, \dots, \boldsymbol{x_n}\}, \|.\|_p, \alpha_p\right)$. By definition of a covering number, there exists $C = \{\boldsymbol{c_1}, \dots, \boldsymbol{c_N}\}$ an α_p-covering of $\{\boldsymbol{x_1}, \dots \boldsymbol{x_n}\}$ for the ℓ_p norm. Furthermore, for $j \in \{1, \dots, N\}$ and $y \in \{1, \dots, K\}$, we define*

$$E_{y,j} = \left\{ i \in \{1, \dots, n\} \text{ s.t. } y_i = y \text{ and } \underset{l \in \{1,\dots,N\}}{\operatorname{argmin}} \|x_i - c_l\| = j \right\}.$$

We also denote $E_j = \underset{y \in [K]}{\cup} E_{y,j}$. Finally, we denote $\mathcal{L}_{\mathrm{m}} : (\boldsymbol{x}, y) \mapsto \mathcal{L}_{0/1}(\mathrm{m}(\boldsymbol{x}), y)$. Then, by definition of the empirical Rademacher complexity, we can write

$$\mathfrak{R}_{\mathcal{S}}\left(\mathcal{L}_{\mathcal{M}_{TV}(\alpha_p,\epsilon)}\right) = \frac{1}{n}\mathbb{E}_{r_i}\left[\sup_{\mathrm{m} \in \mathcal{M}_{TV}(\alpha_p,\epsilon)} \sum_{i=1}^{n} r_i \mathcal{L}_{\mathrm{m}}(\boldsymbol{x_i}, y_i)\right].$$

Then we can use E_j to write

$$\mathfrak{R}_{\mathcal{S}}\left(\mathcal{L}_{\mathcal{M}_{TV}(\alpha_p,\epsilon)}\right) = \frac{1}{n}\mathbb{E}_{r_i}\left[\sup_{\mathrm{m} \in \mathcal{M}_{TV}(\alpha_p,\epsilon)} \sum_{j=1}^{N} \sum_{i \in E_j} r_i \mathcal{L}_{\mathrm{m}}(\boldsymbol{x_i}, y_i)\right].$$

Furthermore for any $\mathrm{m} \in \mathcal{M}_{TV}(\alpha_p, \epsilon)$ and $i \in E_j$, there exists $\epsilon_i \in [-\epsilon, \epsilon]$ such that: $\mathcal{L}_{\mathrm{m}}(\boldsymbol{x_i}, y_i) = \mathcal{L}_{\mathrm{m}}(\boldsymbol{c_j}, y_i) + \epsilon_i$. Then we have

$$\begin{aligned}\mathfrak{R}_{\mathcal{S}}\left(\mathcal{L}_{\mathcal{M}_{TV}(\alpha_p,\epsilon)}\right) \leq\ & \frac{1}{n}\mathbb{E}_{r_i}\left[\sup_{\mathrm{m} \in \mathcal{M}_{TV}(\alpha_p,\epsilon)} \sum_{j=1}^{N} \sum_{i \in E_j} r_i \mathcal{L}_{\mathrm{m}}(\boldsymbol{c_j}, y_i)\right] \\ & + \frac{1}{n}\mathbb{E}_{r_i}\left[\sup_{\epsilon_i \in [-\epsilon,\epsilon]} \sum_{j=1}^{N} \sum_{i \in E_j} r_i \epsilon_i\right].\end{aligned}$$

Let us start by studying the second term. We have

$$\frac{1}{n}\mathbb{E}_{r_i}\left[\sup_{\epsilon_i \in [-\epsilon,\epsilon]} \sum_{j=1}^{N} \sum_{i \in E_j} r_i \epsilon_i\right] = \frac{1}{n}\mathbb{E}_{r_i}\left[\sup_{\epsilon_i \in [-\epsilon,\epsilon]} \sum_{i=1}^{n} r_i \epsilon_i\right] = \frac{1}{n}\sum_{i=1}^{n} \epsilon = \epsilon.$$

Now looking at the first term. Since $\mathcal{L}_{\mathrm{m}}(\boldsymbol{x}, y) \in [0,1]$ *for all* $(\boldsymbol{x}, y)$ *we have*

$$\begin{aligned}\frac{1}{n}\mathbb{E}_{r_i}\left[\sup_{\mathrm{m}\in\mathcal{M}_{TV}(\alpha_p,\epsilon)}\sum_{j=1}^{N}\sum_{i\in E_j} r_i\mathcal{L}_{\mathrm{m}}(\boldsymbol{c_j}, y_i)\right] &= \frac{1}{n}\mathbb{E}_{r_i}\left[\sup_{\mathrm{m}\in\mathcal{M}_{TV}(\alpha_p,\epsilon)}\sum_{j=1}^{N}\sum_{y=1}^{K}\mathcal{L}_{\mathrm{m}}(\boldsymbol{c_j}, y)\sum_{i\in E_{y,j}} r_i\right] \\ &\leq \frac{1}{n}\mathbb{E}_{r_i}\left[\sum_{j=1}^{N}\sum_{y=1}^{K}\left|\sum_{i\in E_{y,j}} r_i\right|\right].\end{aligned}$$

Finally using the Khintchine inequality and the Cauchy Schartz inequality we get

$$\begin{aligned}\frac{1}{n}\mathbb{E}_{r_i}\left[\sum_{j=1}^{N}\sum_{y=1}^{K}\left|\sum_{i\in E_{y,j}} r_i\right|\right] &\leq \frac{1}{n}\sum_{j=1}^{N}\sum_{y=1}^{K}\sqrt{|E_{y,j}|} \quad \textit{(Khintchine)} \\ &\leq \frac{1}{n}\sqrt{N\times K}\sqrt{\sum_{j=1}^{N}\sum_{y=1}^{K}|E_{y,j}|} \quad \textit{(Cauchy)} \\ &= \sqrt{\frac{N\times K}{n}}.\end{aligned}$$

By combining the upper-bounds we have for each term, we get the expected result,

$$\mathfrak{R}_{\mathcal{S}}\left(\mathcal{L}_{\mathcal{M}_{TV}(\alpha_p,\epsilon)}\right) \leq \sqrt{\frac{N\times K}{n}} + \epsilon.$$

The above result means that, if we can cover the n training samples with $O(1)$ balls, then we can bound the generalization gap of any randomized classifier $\mathrm{m} \in \mathcal{M}_{TV}(\alpha_p, \epsilon)$ by $O\left(\frac{1}{\sqrt{n}}\right) + \epsilon$. Furthermore, a natural corollary of Theorem 5 bounds the Rademacher complexity of the class $\mathcal{L}_{\mathcal{M}_\beta(\alpha_p,\epsilon)}$.

Corollary 1 *Let* $\mathcal{L}_{\mathcal{M}_\beta(\alpha_p,\epsilon)}$ *be the loss function class associated with* $\mathcal{M}_\beta(\alpha_p, \epsilon)$*. Then, for any* $\mathcal{S} := \{(\boldsymbol{x_1}, y_1), \ldots, (\boldsymbol{x_n}, y_n)\}$*, the following holds,*

$$\mathfrak{R}_{\mathcal{S}}\left(\mathcal{L}_{\mathcal{M}_\beta(\alpha_p,\epsilon)}\right) \leq \sqrt{\frac{N\times K}{n}} + \min\left(\frac{3}{2}\left(\sqrt{1+\frac{4\epsilon}{9}}-1\right)^{1/2}, \frac{e^{\epsilon+1}-1}{e^{\epsilon+1}+1}\right).$$

Where $N = N\left(\{\boldsymbol{x_1}, \ldots, \boldsymbol{x_n}\}, \|.\|_p, \alpha_p\right)$ *is the* α_p*-external covering number of the inputs* $\{\boldsymbol{x_1}, \ldots, \boldsymbol{x_n}\}$ *for the* ℓ_p *norm.*

Proof 6 *This corollary is an immediate consequence of Theorem 5 and Proposition 2.*

Thanks to Theorems 4 and 5 and Corollary 1, one can easily bound the generalization gap of robust randomized classifiers.

5.2 Discussion and dimensionality issues

[60] previously studied generalization bounds for learning algorithms based on their robustness. Although we use very different proof techniques, their results and ours are similar. More precisely, both analyses conclude that robust models generalize well if the training samples have a small covering number. Note, however, that we base our formulation on an *adaptive partition* of the samples, while the initial paper from [60] only focuses on a fixed partition of the input space. The interested reader can refer to the discussion section in [60] for more details.

These findings seem to contradict the current line of works on the hardness of generalization in the adversarial setting. In fact, if the ground truth distribution is sufficiently concentrated (*e.g.* lies in a low dimensional subspace of $\boldsymbol{x}$), a small number of balls can cover $\mathcal{S}$ with high probability; hence $N = O(1)$. This means that we can learn robust classifiers with the same sample complexity as in the standard setting. But if the ground truth distribution is not concentrated enough, the training samples will be far one from another; hence forcing the covering number to be large. In the worse case scenario, we need to cover the whole space $[0,1]^d$ giving a covering number $N = O\left(\frac{1}{(\alpha_p)^d}\right)$ which is exponential in the dimension of the problem.

Therefore, in the worst-case scenario, our bound is in $O\left(\frac{1}{(\alpha_p)^d\sqrt{n}}\right) + \epsilon$. When α_p is small and the dimension of the problem is high, this bound is too large to give any meaningful insight on the generalization gap of the problem. Therefore, we still need to tighten our analysis to show that robust learning for randomized classifiers is possible in high dimensional spaces.

Remark 3 *Note that, we provided a very general result for randomized classifiers under the only assumption that they are robust* w.r.t. *the total variation distance. Our result applies to any class of classifiers and not only linear classifiers or one-hidden layer neural networks. To build a finer analysis, and to evade the curse of dimensionality, we should consider designing specific sub-classes $\mathcal{M} \subset \mathcal{M}_{TV}(\alpha_p, \epsilon)$ and adapt the proofs to make the term N smaller in the worst-case scenario.*

6 Building robust randomized classifiers

In this section we present a simple yet efficient way to transform a non-robust, non-randomized classifier into a robust randomized classifier. To do so, we use a key property of both the Renyi divergence and the total variation distance called the *Data processing inequality*. It is a well-known result from information theory which states that *"post-processing cannot increase information"*. The data processing inequality is as follows.

Theorem 6 ([10]) *Let us consider two arbitrary spaces $\mathcal{Z}, \mathcal{Z}'$, $\rho, \rho' \in \mathcal{P}(\mathcal{Z})$ and $D \in \{D_{TV}, D_\beta\}$. Then for any $\psi : \mathcal{Z} \to \mathcal{Z}'$ we have*

$$D(\psi\#\rho, \psi\#\rho') \leq D(\rho, \rho'),$$

where $\psi\#\rho$ denotes the pushforward of distiburtion ρ by ψ.

In the context of robustness to adversarial examples, we use the data processing inequality to ease the design of robust randomized classifiers. In particular, let us suppose that we can build a randomized pre-processing $\mathfrak{p} : \mathcal{X} \to \mathcal{P}(\mathcal{X})$ such that for any $\boldsymbol{x} \in \mathcal{X}$ and any α_p-bounded perturbation $\boldsymbol{\tau}$, we have

$$D(\mathfrak{p}(\boldsymbol{x}), \mathfrak{p}(\boldsymbol{x} + \boldsymbol{\tau})) \leq \epsilon, \text{ with } D \in \{D_{TV}, D_\beta\}. \tag{13}$$

Then, thanks to the data processing inequality, we can take any deterministic classifier $\boldsymbol{h}$ to build an (α_p, ϵ) robust classifier w.r.t D defined as $\mathrm{m} : \boldsymbol{x} \mapsto \boldsymbol{h}\#\mathfrak{p}(\boldsymbol{x})$. This considerably simplifies the problem of building a class of robust models. Therefore, we want to build $\mathfrak{p}$ a randomized pre-processing for which we can control the Renyi divergence and/or total variation distance between two inputs. To do this, we analyze the simple procedure of injecting random noise directly on the image before sending it to a classifier. Since the Renyi divergence and the total variation distances are particularly well suited to the study of Gaussian distributions, we first use this type of noise injection. More precisely, in this section, we focus on a mapping that writes as follows.

$$\mathfrak{p} : \boldsymbol{x} \mapsto \mathcal{N}(\boldsymbol{x}, \Sigma), \tag{14}$$

for some given non-degenerate covariance matrix $\Sigma \in \mathcal{M}_{d\times d}(\mathbb{R})$. We refer the interested reader to [39] for more general classes of noise, namely exponential families. Let us now evaluate the maximal variation of Gaussian pre-processing $\mathfrak{p}$ when applied to an image $\boldsymbol{x} \in \mathcal{X}$ with and without perturbation.

Lemma 1 *Let $\beta > 1$, $\boldsymbol{x}, \boldsymbol{\tau} \in \mathcal{X}$ and $\Sigma \in \mathcal{M}_{d\times d}(\mathbb{R})$ a non-degenerate covariance matrix. Let $\rho = \mathcal{N}(\boldsymbol{x}, \Sigma)$ and $\rho' = \mathcal{N}(\boldsymbol{x}+\boldsymbol{\tau}, \Sigma)$, then $D_\beta(\rho, \rho') = \frac{\beta}{2}\|\boldsymbol{\tau}\|^2_{\Sigma^{-1}}$.*

Proof 7 *Let $\beta > 1$. Let us denote g and g' respectively the probability density functions of ρ and ρ' with respect to the Lebesgue measure. We also set $\boldsymbol{x}' = \boldsymbol{x} + \boldsymbol{\tau}$ for readability. Then we have*

$$\begin{aligned} D_\beta(\rho, \rho') &= \frac{1}{\beta - 1} \log \mathbb{E}_{\boldsymbol{z}\sim\rho'} \left[\left(\frac{g(\boldsymbol{z})}{g'(\boldsymbol{z})} \right)^\beta \right] \\ &= \frac{1}{\beta - 1} \log \mathbb{E}_{\boldsymbol{z}\sim\rho'} \Big[\exp\Big(\frac{\beta}{2} \big((\boldsymbol{z}-\boldsymbol{x}')^\intercal \Sigma^{-1} (\boldsymbol{z}-\boldsymbol{x}') - (\boldsymbol{z}-\boldsymbol{x})^\intercal \Sigma^{-1} (\boldsymbol{z}-\boldsymbol{x}) \big) \Big) \Big]. \end{aligned}$$

By change of variable we get

$$\begin{aligned} &= \frac{1}{\beta - 1} \log \mathbb{E}_{\boldsymbol{z}\sim\mathcal{N}(0,\Sigma)} \left[\exp\left(\frac{\beta}{2} \big(\boldsymbol{z}^\intercal \Sigma^{-1} \boldsymbol{z} - (\boldsymbol{z}+\boldsymbol{\tau})^\intercal \Sigma^{-1} (\boldsymbol{z}+\boldsymbol{\tau}) \big) \right) \right] \\ &= \frac{1}{\beta - 1} \log \mathbb{E}_{\boldsymbol{z}\sim\mathcal{N}(0,\Sigma)} \left[\exp\left(\frac{\beta}{2} \left(-2\boldsymbol{z}^\intercal \Sigma^{-1} \boldsymbol{\tau} - \|\boldsymbol{\tau}\|^2_{\Sigma^{-1}} \right) \right) \right] \\ &= \frac{1}{\beta - 1} \log \int_{\mathbb{R}^d} \frac{\exp\left(-\frac{1}{2}\boldsymbol{z}^\intercal \Sigma^{-1} \boldsymbol{z} - \frac{\beta}{2} 2\boldsymbol{z}^\intercal \Sigma^{-1} \boldsymbol{\tau} - \frac{\beta}{2}\|\boldsymbol{\tau}\|^2_{\Sigma^{-1}} \right)}{(2\pi)^d \det(\Sigma)^{d/2}} d\boldsymbol{z}\,. \end{aligned}$$

Furthermore, for any $\boldsymbol{z} \in \mathbb{R}^d$, we have

$$\begin{aligned} &-\frac{1}{2}\boldsymbol{z}^\intercal \Sigma^{-1} \boldsymbol{z} - \frac{\beta}{2} 2\boldsymbol{z}^\intercal \Sigma^{-1} \boldsymbol{\tau} - \frac{\beta}{2}\|\boldsymbol{\tau}\|^2_{\Sigma^{-1}} \\ =& -\frac{1}{2}(\boldsymbol{z}+\beta\boldsymbol{\tau})^\intercal \Sigma^{-1} (\boldsymbol{z}+\beta\boldsymbol{\tau}) + \frac{\beta^2 - \beta}{2}\|\boldsymbol{\tau}\|^2_{\Sigma^{-1}}\,. \end{aligned}$$

Then we can re-write the Renyi divergence as follows

$$\begin{aligned} D_\beta(\rho, \rho') &= \frac{1}{\beta - 1} \log \mathbb{E}_{\boldsymbol{z}\sim\mathcal{N}(-\beta\boldsymbol{\tau},\Sigma)} \left[\exp\left(\frac{\beta^2 - \beta}{2}\|\boldsymbol{\tau}\|^2_{\Sigma^{-1}} \right) \right] \\ &= \frac{1}{\beta - 1} \log \left(\exp\left(\frac{\beta^2 - \beta}{2}\|\boldsymbol{\tau}\|^2_{\Sigma^{-1}} \right) \right) \\ &= \frac{\beta}{2}\|\boldsymbol{\tau}\|^2_{\Sigma^{-1}}\,. \end{aligned}$$

This concludes the proof.

Thanks to the above lemma, we know how to evaluate the level of Renyi-robustness that a Gaussian noise pre-processing brings to a classifier. Now that we have this result, thanks to Proposition 2, we can also upper-bound the total variation distance between $\mathcal{N}(\boldsymbol{x}, \Sigma)$ and $\mathcal{N}(\boldsymbol{x}+\boldsymbol{\tau}, \Sigma)$. But this bound is not always tight. Besides, we can directly evaluate the total variation distance between two Gaussian distributions as follows.

Lemma 2 *Let $\boldsymbol{x}, \boldsymbol{x}' \in \mathcal{X}$ and $\Sigma \in \mathcal{M}_{d\times d}(\mathbb{R})$ a non-degenerate covariance matrix. Let $\rho = \mathcal{N}(\boldsymbol{x}, \Sigma)$ and $\rho' = \mathcal{N}(\boldsymbol{x}+\boldsymbol{\tau}, \Sigma)$, then $D_{TV}(\rho, \rho') = 2\Phi(\frac{\|\boldsymbol{\tau}\|_{\Sigma^{-1}}}{2}) - 1$ with Φ the cumulative density function of the standard Gaussian distribution.*

Proof 8 *Let us denote g and g' respectively the probability density functions of ρ and ρ' with respect to the Lebesgue measure. Furthermore, we denote $\boldsymbol{x}' = \boldsymbol{x} + \boldsymbol{\tau}$. Then by definition of the total variation distance, we have $D_{TV}(\rho, \rho) = \rho(Z) - \rho'(Z)$ with $Z = \{\boldsymbol{z} \text{ s.t. } g(\boldsymbol{z}) \geq g'(\boldsymbol{z})\}$. In our case $g(\boldsymbol{z}) \geq g'(\boldsymbol{z})$ is equivalent to*

$$(\boldsymbol{z}-\boldsymbol{x}')^\intercal \Sigma^{-1} (\boldsymbol{z}-\boldsymbol{x}') - (\boldsymbol{z}-\boldsymbol{x})^\intercal \Sigma^{-1} (\boldsymbol{z}-\boldsymbol{x}) \geq 0.$$

Then with the same simplification as above, we have

$$\begin{aligned}\rho(Z) &= \mathbb{P}_{\boldsymbol{z}\sim\mathcal{N}(\boldsymbol{x},\Sigma)}\left((\boldsymbol{z}-\boldsymbol{x}')^\intercal\Sigma^{-1}(\boldsymbol{z}-\boldsymbol{x}')-(\boldsymbol{z}-\boldsymbol{x})^\intercal\Sigma^{-1}(\boldsymbol{z}-\boldsymbol{x})\geq 0\right)\\ &= \mathbb{P}_{\boldsymbol{z}\sim\mathcal{N}(0,\Sigma)}\left((\boldsymbol{z}-\boldsymbol{\tau})^\intercal\Sigma^{-1}(\boldsymbol{z}-\boldsymbol{\tau})-\boldsymbol{z}^\intercal\Sigma^{-1}\boldsymbol{z}\geq 0\right)\\ &= \mathbb{P}_{\boldsymbol{z}\sim\mathcal{N}(0,\Sigma)}\left(-2\boldsymbol{z}^\intercal\Sigma^{-1}\boldsymbol{\tau}+\|\boldsymbol{\tau}\|^2_{\Sigma^{-1}}\geq 0\right)\\ &= \mathbb{P}_{\boldsymbol{z}\sim\mathcal{N}(0,I_d)}\left(\boldsymbol{z}^\intercal\Sigma^{-1/2}\boldsymbol{\tau}\leq\frac{1}{2}\|\boldsymbol{\tau}\|^2_{\Sigma^{-1}}\right).\end{aligned}$$

Furthermore, if $\boldsymbol{z}\sim\mathcal{N}(0,I_d)$ *then* $\boldsymbol{z}^\intercal\Sigma^{-1/2}\boldsymbol{\tau}\sim\mathcal{N}(0,\|\boldsymbol{\tau}\|^2_{\Sigma^{-1}})$*; hence we also have* $\frac{\boldsymbol{z}^\intercal\Sigma^{-1/2}\boldsymbol{\tau}}{\|\boldsymbol{\tau}\|_{\Sigma^{-1}}}\sim\mathcal{N}(0,1)$*. Accordingly we get*

$$\rho(Z)=\mathbb{P}_{\boldsymbol{z}\sim\mathcal{N}(0,1)}\left(\boldsymbol{z}\leq\frac{1}{2}\|\boldsymbol{\tau}\|_{\Sigma^{-1}}\right)=\Phi\left(\frac{1}{2}\|\boldsymbol{\tau}\|_{\Sigma^{-1}}\right).$$

By symmetry we get that $\rho'(A)=1-\rho(A)=1-\Phi\left(\frac{1}{2}\|\boldsymbol{\tau}\|_{\Sigma^{-1}}\right)$*. We then get*

$$D_{TV}(\mu,\nu)=2\Phi\left(\frac{\|\boldsymbol{\tau}\|_{\Sigma^{-1}}}{2}\right)-1$$

which concludes the proof.

Note that both bounds increase with the Mahalanobis norm of $\boldsymbol{\tau}$. Furthermore, we see that the greater the entropy of the Gaussian noise we inject, the smaller the distance between distributions. If we simplify the covariance matrix by setting $\Sigma=\sigma^2 I_d$, it means that we can build more or less robust randomized classifiers against ℓ_2 adversaries, depending on σ.

Theorem 7 (Robustness of Gaussian pre-processing) *Let us consider* $c:\mathcal{X}\to\mathcal{Y}$ *a deterministic classifier,* $\sigma>0$ *and* $\mathfrak{p}:\boldsymbol{x}\mapsto\mathcal{N}(\boldsymbol{x},\sigma^2 I_d)$ *a pre-processing probabilistic mapping. Then the randomized classifier* $\mathrm{m}:=c\#\mathfrak{p}$ *is*

- $(\alpha_2,\frac{(\alpha_2)^2\beta}{2\sigma})$*-robust* w.r.t. D_β *against* ℓ_2 *adversaries.*
- $(\alpha_2,\ 2\Phi\left(\frac{\alpha_2}{2\sigma}\right)-1)$*-robust* w.r.t. D_{TV} *against* ℓ_2 *adversaries.*

Proof 9 *Let* $\boldsymbol{x},\boldsymbol{\tau}\in\mathcal{X}$ *such that* $\|\boldsymbol{\tau}\|_2\leq\alpha_2$*. Thanks to Lemma 1 we have*

$$D_\beta(\mathfrak{p}(\boldsymbol{x}),\mathfrak{p}(\boldsymbol{x}+\boldsymbol{\tau}))=\frac{\beta}{2}\|\boldsymbol{\tau}\|^2_{\Sigma^{-1}}=\frac{\beta}{2\sigma^2}\|\boldsymbol{\tau}\|_2^2\leq\frac{\beta(\alpha_2)^2}{2\sigma^2}.$$

Similarly, thanks to Lemma 2, we get

$$D_{TV}(\mathfrak{p}(\boldsymbol{x}),\mathfrak{p}(\boldsymbol{x}+\boldsymbol{\tau}))=2\Phi\left(\frac{\|\boldsymbol{\tau}\|_{\Sigma^{-1}}}{2}\right)-1\leq 2\Phi\left(\frac{\alpha_2}{2\sigma}\right)-1.$$

Finally, from the data processing inequality, i.e. *Theorem 6, we get both*

$$D_\beta(\mathrm{m}(\boldsymbol{x}),\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau}))\leq\frac{\beta(\alpha_2)^2}{2\sigma^2},$$

and

$$D_{TV}(\mathrm{m}(\boldsymbol{x}),\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau}))\leq 2\Phi\left(\frac{\alpha_2}{2\sigma}\right)-1.$$

The above inequalities conclude the proof.

Theorem 7 means that we can build simple noise injection schemes as pre-processing of state-of-the-art image classification models and keep track of the maximal loss of accuracy under attack of the resulting randomized classifier. These results also highlight the profound link between randomized classifiers and randomized smoothing as presented by [9]. Even though our findings are of different nature, both techniques use the same base mechanism (Gaussian noise injection). Therefore, Gaussian pre-processing is a principled defense method that can be analyzed through several standpoints, including certified robustness and statistical learning theory.

7 Discussion: Mode preservation and Randomized Smoothing

Even though randomized classifiers have some interesting properties regarding generalization error, we can also study them through the prism of deterministic robustness. Let us for example consider the classifier that outputs the class with the highest probability for $\mathrm{m}(\boldsymbol{x})$, *a.k.a.* the mode of $\mathrm{m}(\boldsymbol{x})$. It writes

$$\boldsymbol{h}_{\text{rob}} : \boldsymbol{x} \mapsto \underset{k\in[K]}{\operatorname{argmax}}\, \mathrm{m}(\boldsymbol{x})_k \tag{15}$$

Then checking whether $\boldsymbol{h}_{\text{rob}}$ is robust boils down to demonstrating that the mode of $\mathrm{m}(\boldsymbol{x})$ does not change under perturbation. It turns out that D_{TV} robust classifiers have this property. We call it the mode preservation property of $\mathcal{M}_{TV}(\alpha_p, \epsilon)$.

Proposition 3 (Mode preservation for D_{TV}-robust classifiers) *Let $\mathrm{m} \in \mathcal{M}_{TV}(\alpha_p, \epsilon)$ be a robust randomized classifier and $\boldsymbol{x} \in \mathcal{X}$ such that $\mathrm{m}(\boldsymbol{x})_{(1)} \geq \mathrm{m}(\boldsymbol{x})_{(2)} + 2\epsilon$. Then, for any $\boldsymbol{\tau} \in \mathcal{X}$, the following holds,*

$$\|\boldsymbol{\tau}\|_p \leq \alpha_p \implies \boldsymbol{h}_{rob}(\boldsymbol{x}) = \boldsymbol{h}_{rob}(\boldsymbol{x} + \boldsymbol{\tau}) \ .$$

Proof 10 *Let $\boldsymbol{x}, \boldsymbol{\tau} \in \mathcal{X}$ such that $\|\boldsymbol{\tau}\|_p \leq \alpha_p$ and $\mathrm{m} \in \mathcal{M}_{TV}(\alpha_p, \epsilon)$ such that*

$$\mathrm{m}(\boldsymbol{x})_{(1)} \geq \mathrm{m}(\boldsymbol{x})_{(2)} + 2\epsilon.$$

By definition of $\mathcal{M}_{TV}(\alpha_p, \epsilon)$, we have that

$$D_{TV}(\mathrm{m}(\boldsymbol{x}), \mathrm{m}(\boldsymbol{x} + \boldsymbol{\tau})) \leq \epsilon.$$

Then, for all $k \in \{1, \ldots, K\}$ we have

$$\mathrm{m}(\boldsymbol{x})_k - \epsilon \leq \mathrm{m}(\boldsymbol{x} + \boldsymbol{\tau})_k \leq \mathrm{m}(\boldsymbol{x})_k + \epsilon \, .$$

Let us denote k^* the index of the biggest value in $\mathrm{m}(\boldsymbol{x})$, i.e. *$\mathrm{m}(\boldsymbol{x})_{k^*} = \mathrm{m}(\boldsymbol{x})_{(1)}$. For any $k \in \{1, \ldots, K\}$ with $k \neq k^*$, we have $\mathrm{m}(\boldsymbol{x})_{k^*} \geq \mathrm{m}(\boldsymbol{x})_k + 2\epsilon$. Finally, for any $k \neq k^*$, we get*

$$\mathrm{m}(\boldsymbol{x} + \boldsymbol{\tau})_{k^*} \geq \mathrm{m}(\boldsymbol{x})_{k^*} - \epsilon \geq \mathrm{m}(\boldsymbol{x})_k + \epsilon \geq \mathrm{m}(\boldsymbol{x} + \boldsymbol{\tau})_k.$$

Then, $\underset{k\in[K]}{\operatorname{argmax}}\, \mathrm{m}(\boldsymbol{x})_k = \underset{k\in[K]}{\operatorname{argmax}}\, \mathrm{m}(\boldsymbol{x} + \boldsymbol{\tau})_k$. This concludes the proof.

Similarly, we can demonstrate a mode preservation property for robust classifiers w.r.t. the Renyi divergence.

Proposition 4 (Mode preservation for Renyi-robust classifiers) *Let $\mathrm{m} \in \mathcal{M}_\beta(\alpha_p, \epsilon)$ be a robust randomized classifier and $\boldsymbol{x} \in \mathcal{X}$ such that $\left(\mathrm{m}(\boldsymbol{x})_{(1)}\right)^{\frac{\beta}{\beta-1}} \geq \exp\left((2 - \frac{1}{\beta})\epsilon\right)\left(\mathrm{m}(\boldsymbol{x})_{(2)}\right)^{\frac{\beta-1}{\beta}}$. Then, for any $\boldsymbol{\tau} \in \mathcal{X}$, the following holds,*

$$\|\boldsymbol{\tau}\|_p \leq \alpha_p \implies \boldsymbol{h}_{rob}(\boldsymbol{x}) = \boldsymbol{h}_{rob}(\boldsymbol{x} + \boldsymbol{\tau}),$$

where $\boldsymbol{h}_{rob}(\boldsymbol{x}) := \underset{k\in[K]}{\operatorname{argmax}}\, \mathrm{m}(\boldsymbol{x})_k$.

Proof 11 *Let $\boldsymbol{x}, \boldsymbol{\tau} \in \mathcal{X}$ such that $\|\boldsymbol{\tau}\|_p \leq \alpha_p$ and $\mathrm{m} \in \mathcal{M}_\beta(\alpha_p, \epsilon)$ such that*

$$\left(\mathrm{m}(\boldsymbol{x})_{(1)}\right)^{\frac{\beta}{\beta-1}} \geq \exp\left((2 - \frac{1}{\beta})\epsilon\right)\left(\mathrm{m}(\boldsymbol{x})_{(2)}\right)^{\frac{\beta-1}{\beta}} .$$

Then by definition of $\mathcal{M}_\beta(\alpha_p, \epsilon)$, we have

$$D_\beta(\mathrm{m}(\boldsymbol{x}), \mathrm{m}(\boldsymbol{x} + \boldsymbol{\tau})) \leq \epsilon.$$

Furthermore, by using Proposition 1, for any $k \in \{1, \dots, K\}$ *we have*

$$(*)\, \mathrm{m}(\boldsymbol{x})_k \leq \left(\exp(\epsilon)\, \mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau})_k\right)^{\frac{\beta-1}{\beta}} \text{ and } (**)\, \mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau})_k \leq \left(\exp(\epsilon)\, \mathrm{m}(\boldsymbol{x})_k\right)^{\frac{\beta-1}{\beta}} .$$

Let us denote k^* *the index such that* $\mathrm{m}(\boldsymbol{x})_{k^*} = \mathrm{m}(\boldsymbol{x})_{(1)}$. *Then using* $(*)$ *we get*

$$\mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau})_{k^*} \geq \exp(-\epsilon)(\mathrm{m}(\boldsymbol{x})_{k^*})^{\frac{\beta}{\beta-1}}.$$

Furthermore for any $k \in \{1, \dots, K\}$ *where* $k \neq k^*$, *we can use the assumption we made on* m *to get*

$$\exp(-\epsilon)(\mathrm{m}(\boldsymbol{x})_{k^*})^{\frac{\beta}{\beta-1}} \geq \exp\left(\frac{\beta-1}{\beta}\epsilon\right)(\mathrm{m}(\boldsymbol{x})_k)^{\frac{\beta-1}{\beta}}.$$

Finally, using $(**)$ *we have*

$$\exp\left(\frac{\beta-1}{\beta}\epsilon\right)(\mathrm{m}(\boldsymbol{x})_k)^{\frac{\beta-1}{\beta}} \geq \mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau})_k.$$

The above gives us $\underset{k\in[K]}{\operatorname{argmax}}\, \mathrm{m}(\boldsymbol{x})_k = \underset{k\in[K]}{\operatorname{argmax}}\, \mathrm{m}(\boldsymbol{x}+\boldsymbol{\tau})_k$. *This concludes the proof.*

Coming back to the decomposition in Equation (5), with the above result, we can bound the risk the adversary induces with non-zero perturbations by the mass of points on which the classifier $\boldsymbol{h}_{\mathrm{rob}}$ gives the good response but based on a low probability of success, *i.e.* with small confidence

$$\mathcal{R}^{\mathrm{adv}}_{>0}(\mathrm{m}) \leq \mathbb{P}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[\boldsymbol{h}_{\mathrm{rob}}(\boldsymbol{x}) = y \text{ and } \mathrm{m}(\boldsymbol{x})_{(1)} < \mathrm{m}(\boldsymbol{x})_{(2)} + 2\epsilon\right]. \tag{16}$$

This means that the only points on which the adversary may induce misclassification are the points on which m already has a high risk. Once more, this says something fundamental about the behavior of robust randomized classifiers. On undefended models, the adversary could change the decision on any point it wanted; now it is limited to changing points on which the classifier is already inaccurate. This considerably mitigates the threat model we should consider. Furthermore, for any deterministic classifier designed as in Equation (15), we can also bound the maximal loss of accuracy under attack the classifier may suffer. This bound may, however, be harder to evaluate since it now depends on both the classifier and the dataset distribution. The classifier we define in Equation (15) and the mode preservation property of m are closely related to provable defenses based on randomized smoothing. The core idea of randomized smoothing is to take a hypothesis $\boldsymbol{h}$ and to build a robust classifier that writes

$$c_{rob} : \boldsymbol{x} \mapsto \underset{k\in[K]}{\operatorname{argmax}}\, \mathbb{P}_{\boldsymbol{z}\sim\mathcal{N}(0,\sigma^2 I)}\left[\boldsymbol{h}(\boldsymbol{x}+\boldsymbol{z}) = k\right]. \tag{17}$$

From a probabilistic point of view, for any input $\boldsymbol{x}$, randomized smoothing amounts to output the most probable class of the probability measure $\mathrm{m}(\boldsymbol{x}) := \boldsymbol{h}\#\mathcal{N}\left(\boldsymbol{x}, \sigma^2 I\right)$. Hence, randomized smoothing uses the mode preservation property of m to build a provably robust (deterministic) classifier. Therefore, the above results (Proposition 3 and Equation 16) also hold for provable defenses based on randomized smoothing. Studying randomized smoothing from our point of view could give an interesting new perspective on that method. So far no results have been published on the generalisation gap of this defense in the adversarial setting. We could devise generalization bounds by similarity with our analysis. Furthermore, the probabilistic interpretation stresses that randomized smoothing is somewhat restrictive since it only considers probability measures which are the expectation on a simple noise injection scheme. The mode preservation property explains the behavior of randomized smoothing, but also presents fundamental properties of randomized defenses that could be used to construct more general defense schemes.

8 Numerical validations: Gaussian Noise and ℓ_2 adversary

To illustrate our findings, we train randomized neural networks with Gaussian pre-processing during training and inference on CIFAR-10 and CIFAR-100. Based on this randomized classifier, we study the impact of randomization on the standard accuracy of the network, and observe the theoretical trade-off between accuracy and robustness.

8.1 Architecture and training procedure

All the neural networks we use in this section are WideResNets [63] with 28 layers, a widen factor of 10, a dropout factor of 0.3 and LeakyRelu activation with a 0.1 slope. To train an undefended standard classifier we use the following hyper-parameters.

- *Number of Epochs:* 200
- *Batch size:* 400
- *Loss function:* Cross Entropy Loss
- *Optimizer :* Stochastic gradient descent algorithm with momentum 0.9, weight decay of 2×10^{-4} and a learning rate that decreases during the training as follows:

$$lr = \begin{cases} 0.1 & \text{if } 0 \leq \text{epoch} < 60 \\ 0.02 & \text{if } 60 \leq \text{epoch} < 120 \\ 0.004 & \text{if } 120 \leq \text{epoch} < 160 \\ 0.0008 & \text{if } 160 \leq \text{epoch} < 200. \end{cases}$$

To transform these standard networks into randomized classifiers, we inject noise drawn from Gaussian distributions, each with various standard deviations directly on the image before passing it through the network. Both during training and test, for computational efficiency, we evaluate the performance of the the algorithm over a single run for every images; hence no Monte Carlo estimator is used. However, in practice, the test-time accuracy is stable when evaluated over the entire test dataset.

8.2 Results

Figures 1 and 2 show the accuracy and the minimum level of accuracy under attack of our randomized neural network for several levels of injected noise. We can see (Figure 1) that the precision decreases as the noise intensity grows. In that sense, the noise must be calibrated to preserve both accuracy and robustness against adversarial attacks. This is to be expected, because the greater the entropy of the classifier, the less precise it gets.

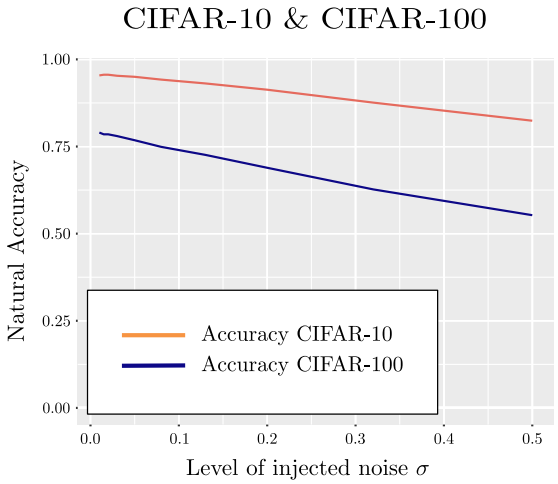

Figure 1: Impact of the standard deviation of the Gausian noise on accuracy in a randomized model on CIFAR-10 and CIFAR-100 dataset.

Furthermore, when injecting Gaussian noise as a defense mechanism, the resulting randomized network m is both $(\alpha_2, \frac{(\alpha_2)^2}{2\sigma})$-robust *w.r.t.* D_1 and $(\alpha_2, 2\Phi\left(\frac{\alpha_2}{2\sigma}\right) - 1)$-robust *w.r.t.* D_{TV} against ℓ_2 adversaries. Therefore thanks to Theorems 1 and 3 we have that

$$\mathcal{R}^{\text{adv}}(\mathrm{m}; \alpha_2) - \mathcal{R}(\mathrm{m}) \leq 2\Phi\left(\frac{\alpha_2}{2\sigma}\right) - 1, \ \textit{and} \tag{18}$$

$$\mathcal{R}^{\text{adv}}(\mathrm{m}; \alpha_2) - \mathcal{R}(\mathrm{m}) \leq 1 - e^{-\frac{(\alpha_2)^2}{2\sigma}} \mathbb{E}_{\boldsymbol{x} \sim \mathcal{D}_{|\mathcal{X}}} \left[e^{-H(\mathrm{m}(\boldsymbol{x}))} \right]. \tag{19}$$

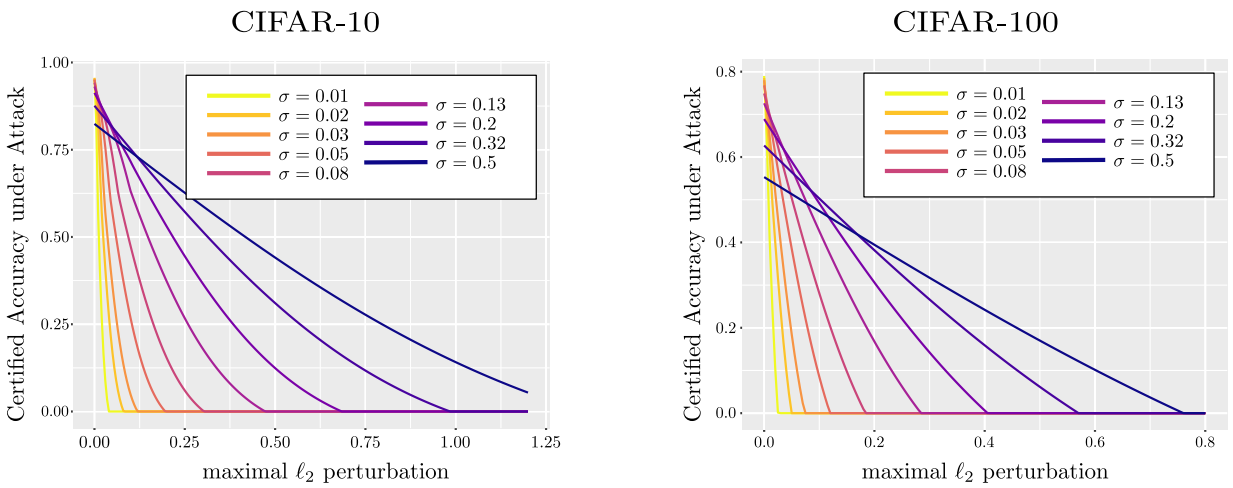

Figure 2: Guaranteed accuracy of different randomized models with Gaussian noise given the ℓ_2 norm of the adversarial perturbations.

Figure 2 illustrates the theoretical lower bound on accuracy under attack (based on the minimum gap between Equations (18) and (19)) for different standard deviations. The term in entropy has been estimated using a Monte Carlo method with 10^4 simulations. The trade-off between accuracy and robustness appears with respect to the noise intensity. With small noises, the accuracy is high, but the guaranteed accuracy drops fast with respect to the magnitude of the adversarial perturbation. Conversely, with bigger noises, the accuracy is lower but decreases slowly with respect to the magnitude of the adversarial perturbation. Overall, we get strong accuracy guarantees against small adversarial perturbations, but when the perturbation is bigger than 0.5 on CIFAR-10 (resp. 0.3 on CIFAR-100, the guarantees are still not sufficient).

9 Lesson learned and future work

This paper brings new contributions to the theory of robustness to adversarial attacks. We provided an in depth analysis of randomized classifier, demonstrating their interest to defend against adversarial attacks. We first defined a notion of robustness for randomized classifiers using probability metrics/divergences, namely the total variation distance and the Renyi divergence. Second, we demonstrated that when a randomized classifier complies with this definition of robustness, we can bound their loss of accuracy under attack. We also studied the generalization properties of this class of functions and gave results indicating that robust randomized classifiers can generalize. Finally, we showed that randomized classifiers have a mode preservation property. This presents a fundamental property of randomized defenses that can be used to explain randomized smoothing from a probabilistic point of view. To support our theoretical findings we presented a simple yet efficient scheme for building robust randomized classifiers. We show that Gaussian noise injection can provide principled robustness against ℓ_2 adversarial attacks. We ran a set of experiments

on CIFAR-10 and CIFAR-100 using Gaussian noise injection with advanced neural network architectures to build accurate models with controlled loss of accuracy under attack.

Future work will focus on studying the combination of randomization with more sophisticated defenses and on devising new tight bounds on the adversarial generalization and the adversarial risk gap of randomized classifiers. Based on the connections we established we randomized smoothing in Section 7, we will also aim at devising bounds on the gap between the standard and adversarial risks for this defense. Another interesting direction would be to show that the classifiers based on randomized smoothing have a generalization gap similar to the classes of randomized classifiers we studied.

A Discussion on the metric/divergence one should consider

As mentioned earlier in this paper, the choice of the metric/divergence is crucial as it characterizes the notion of adversarial robustness we are examining. We focus on the total variation distance and Renyi divergence, but the question of whether these metrics/divergences are more appropriate than others remains open. It should be noted, however, that our definition of robustness is monotonous depending on the metric/divergence we use.

Proposition 5 (Monotonicity of the robustness) *Let* m *be a randomized classifier, and let* D *and* D' *be two divergences/metrics on* $\mathcal{P}(\mathcal{Y})$*. If there exists a non decreasing function* $f : \mathbb{R} \to \mathbb{R}$ *such that* $\forall \rho, \rho' \in \mathcal{P}(\mathcal{Y})$*,* $D(\rho, \rho') \leq f(D'(\rho, \rho'))$*, then the following assertion holds.*

$$\text{m } \textit{is } (\alpha_p, \epsilon)\text{-robust } \textit{w.r.t. } D' \implies \text{m } \textit{is } (\alpha_p, f(\epsilon))\textit{-robust } \text{w.r.t. } D.$$

The proof straightforwardly comes from the definition of robustness.

Proof 12 *Let us consider* m *a randomized classifier* (α_p, ϵ)*-robust* w.r.t. D'*. Then for any* $\boldsymbol{x} \sim \mathcal{D}$*, and* $\boldsymbol{\tau}$ s.t. $\|\boldsymbol{\tau}\|_p \leq \alpha_p$*, since* f *is non decreasing, we have*

$$D(\mathrm{m}(\boldsymbol{x}), \mathrm{m}(\boldsymbol{x} + \boldsymbol{\tau})) \leq f\left(D'(\mathrm{m}(\boldsymbol{x}), \mathrm{m}(\boldsymbol{x} + \boldsymbol{\tau}))\right) \leq f\left(\epsilon\right).$$

Then m *is* $(\alpha_p, f(\epsilon))$*-robust* w.r.t. D *which concludes the proof.*

The above result suggests that the different notions of robustness we might conceive are more related than they appear. Here are some of the most classical divergences used in machine learning. Let ρ, ρ', ν three measures in $\mathcal{P}(\mathcal{Y})$. We denotes g and g' the probability density functions of ρ and ρ' with respect to ν. Then we can define the *Wasserstein distance* as follows

$$D_W(\rho, \rho') := \inf \int_{\mathcal{Y}^2} \mathrm{dist}\left(y, y'\right) d\pi(y, y'), \tag{20}$$

where dist is some ground distance on $\mathcal{Y}$, and the infimum is taken over all joint distributions π in $\mathcal{P}\left(\mathcal{Y} \times \mathcal{Y}\right)$ with marginals ρ and ρ'.

Remark 4 *In transportation theory, the Wasserstein distance is solution of the Monge-Kantorovich problem with the cost function* $c(y, y') = \mathrm{dist}(y, y')$*. Then, the definitions of total variation and Wasserstein distance match when we use the trivial distance* $\mathrm{dist}(y, y') = \mathbb{1}\{y \neq y'\}$*.*

We also define respectively the *Hellinger distance* and the *Separation distance* as follows.

$$D_H(\rho, \rho') := \left[\int_{\mathcal{Y}} \left(\sqrt{g} - \sqrt{g'}\right)^2 d\nu\right]^{1/2}. \tag{21}$$

$$D_S(\rho, \rho') := \sup_{y \in \mathcal{Y}} \left(1 - \frac{g(y)}{g'(y)}\right). \tag{22}$$

If we take any of the above metrics/divergences to instantiate a notion of adversarial robustness we might get very different semantics for them. However, we can show that any of these definitions can be covered – with respect to Proposition 5 – either by the Renyi or the total variation robustness. Figure 3 summarizes the links we can make between all these different definitions of robustness, and Propositions 6 and 7 present the associated results. We can see that the total variation distance and the Renyi divergence are both central since they can cover any of the other robustness notions. This does not mean that they are more appropriate than the others, but at least they are general enough to cover a wide range of possible definitions.

Proposition 6 *Let* m *be a randomized classifier. If* m *is* (α_p, ϵ)*-robust* w.r.t. D_{TV} *then the following assertions hold.*

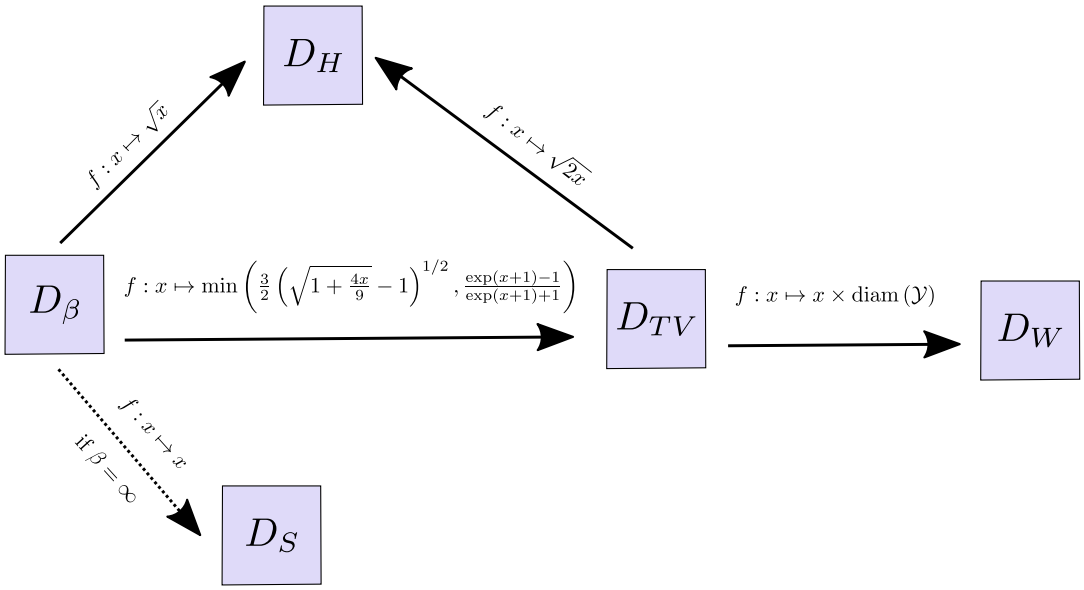

Figure 3: Summary of the relations between the different robustness notions from Propositions 6 and 7.

- m *is* $(\alpha_p, \epsilon \times \mathrm{diam}\,(\mathcal{Y}))$*-robust* w.r.t. D_W*, where* $\mathrm{diam}\,(\mathcal{Y}) := \max_{y,y' \in \mathcal{Y}} \mathrm{dist}(y, y')$.
- m *is* $\left(\alpha_p, \sqrt{2\epsilon}\right)$*-robust* w.r.t. D_H.

Proof 13 *Let us consider* ρ *and* $\rho' \in \mathcal{P}\,(\mathcal{Y})$*. Thanks to [15] we have*

- $D_W(\rho, \rho') \leq \mathrm{diam}(\mathcal{Y}) D_{TV}(\rho, \rho')$.
- $D_H(\rho, \rho') \leq \sqrt{2 D_{TV}(\rho, \rho')}$.

Hence, by using Proposition 5 respectively with $f : x \mapsto \mathrm{diam}(\mathcal{Y})x$ *and* $f : x \mapsto \sqrt{2x}$ *we get the expected results.*

Proposition 7 *Let* m *be a randomized classifier. If* m *is* (α_p, ϵ)*-robust* w.r.t. D_β *then the following assertions hold.*

- m *is* (α_p, ϵ')*-robust* w.r.t. D_{TV} *with* $\epsilon' = \min\left(\frac{3}{2}\left(\sqrt{1+\frac{4\epsilon}{9}}-1\right)^{1/2}, \frac{\exp(\epsilon+1)-1}{\exp(\epsilon+1)+1}\right)$.
- m *is* $(\alpha_p, \sqrt{\epsilon})$*-robust* w.r.t. D_H.
- *If* $\beta = \infty$*, then* m *is* (α_p, ϵ) *robust* w.r.t. D_S.

Proof 14 *1) First, let us suppose that* $\beta \geq 1$*. Thanks to Proposition 2 and to [15], for any* $\rho, \rho' \in \mathcal{P}\,(\mathcal{Y})$ *we have*

- $D_H(\rho, \rho') \leq \sqrt{D_1(\rho, \rho')} \leq \sqrt{D_\beta(\rho, \rho')}$ *(see [15]).*
- $D_{TV}(\rho, \rho') \leq \min\left(\frac{3}{2}\left(\sqrt{1+\frac{4D_\beta(\rho,\rho')}{9}}-1\right)^{1/2}, \frac{\exp\left(D_\beta(\rho,\rho')+1\right)-1}{\exp(D_\beta(\rho,\rho')+1)+1}\right)$ *(Prop. 2).*

Hence, by using Proposition 5, as above, we get the expected results.
2) Now let us suppose that $\beta = \infty$. By definition of the supremum divergence, we have

$$D_\infty(\rho, \rho') = \sup_{B \subset \mathcal{Y}} \left| \ln \frac{\rho(B)}{\rho'(B)} \right|.$$

Furthermore, note that the function $x \mapsto 1 - x - |\ln(x)|$ is negative on $\mathbb{R}$, therefore for any $y \in \mathcal{Y}$ one has

$$1 - \frac{\rho(y)}{\rho'(y)} \leq \left| \ln \frac{\rho(y)}{\rho'(y)} \right|.$$

Since the above inequality is true for any $y \in \mathcal{Y}$, we have

$$D_S(\rho, \rho') = \sup_{y \in \mathcal{Y}} \left(1 - \frac{\rho(y)}{\rho'(y)} \right) \leq \sup_{y \in \mathcal{Y}} \left| \ln \frac{\rho(y)}{\rho'(y)} \right| \leq \sup_{B \subset \mathcal{Y}} \left| \ln \frac{\rho(B)}{\rho'(B)} \right| = D_\infty(\rho, \rho').$$

Finally, by using Proposition 5 with $f : x \mapsto x$ we get the expected results.

References

[1] A. Athalye, N. Carlini, and D. Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018*, July 2018.

[2] P. Awasthi, N. Frank, and M. Mohri. Adversarial learning guarantees for linear hypotheses and neural networks. *International Conference on Machine Learning*, 2020.

[3] P. L. Bartlett and S. Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3:463–482, 2002.

[4] A. Ben-Tal, L. El Ghaoui, and A. Nemirovski. *Robust optimization*, volume 28. Princeton University Press, 2009.

[5] B. Biggio, I. Corona, D. Maiorca, B. Nelson, N. Šrndić, P. Laskov, G. Giacinto, and F. Roli. Evasion attacks against machine learning at test time. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 387–402. Springer, 2013.

[6] N. Carlini and D. Wagner. Adversarial examples are not easily detected: Bypassing ten detection methods. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pages 3–14, 2017.

[7] F. Chapeau-Blondeau and D. Rousseau. Noise-enhanced performance for an optimal bayesian estimator. *IEEE Transactions on Signal Processing*, 52(5):1327–1334, 2004.

[8] P.-Y. Chen, Y. Sharma, H. Zhang, J. Yi, and C.-J. Hsieh. Ead: Elastic-net attacks to deep neural networks via adversarial examples. In *AAAI*, 2018.

[9] J. M. Cohen, E. Rosenfeld, and J. Z. Kolter. Certified adversarial robustness via randomized smoothing. In *International Conference on Machine Learning*, 2019.

[10] T. M. Cover and J. A. Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

[11] F. Croce and M. Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International Conference on Machine Learning*, 2020.

[12] N. Dalvi, P. Domingos, S. Sanghai, and D. Verma. Adversarial classification. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 99–108, 2004.

[13] G. S. Dhillon, K. Azizzadenesheli, J. D. Bernstein, J. Kossaifi, A. Khanna, Z. C. Lipton, and A. Anandkumar. Stochastic activation pruning for robust adversarial defense. In *International Conference on Learning Representations*, 2018.

[14] J. Dong, A. Roth, and W. J. Su. Gaussian differential privacy. *arXiv preprint arXiv:1905.02383*, 2019.

[15] A. L. Gibbs and F. E. Su. On choosing and bounding probability metrics. *International Statistical Review / Revue Internationale de Statistique*, 70(3):419–435, 2002.

[16] G. L. Gilardoni. On pinsker's and vajda's type inequalities for csiszár's f-divergences. *IEEE Transactions on Information Theory*, 56(11):5377–5386, 2010.

[17] A. Globerson and S. Roweis. Nightmare at test time: robust learning by feature deletion. In *Proceedings of the 23rd international conference on Machine learning*, pages 353–360, 2006.

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

[19] Y. Grandvalet, S. Canu, and S. Boucheron. Noise injection: Theoretical prospects. *Neural Computation*, 9(5):1093–1108, 1997.

[20] W. He, J. Wei, X. Chen, N. Carlini, and D. Song. Adversarial example defense: Ensembles of weak defenses are not strong. In *11th {USENIX} Workshop on Offensive Technologies ({WOOT} 17)*, 2017.

[21] S. Hu, T. Yu, C. Guo, W.-L. Chao, and K. Q. Weinberger. A new defense against adversarial images: Turning a weakness into a strength. In *Advances in Neural Information Processing Systems*, pages 1635–1646, 2019.

[22] S. Jetley, N. A. Lord, and P. H. Torr. With friends like these, who needs adversaries? In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 10772–10782, Red Hook, NY, USA, 2018. Curran Associates Inc.

[23] M. Kearns and M. Li. Learning in the presence of malicious errors. *SIAM Journal on Computing*, 22(4):807–837, 1993.

[24] M. J. Kearns, R. E. Schapire, and L. M. Sellie. Toward efficient agnostic learning. *Machine Learning*, 17(2-3):115–141, 1994.

[25] J. Khim and P.-L. Loh. Adversarial risk bounds for binary classification via function transformation. *arXiv preprint arXiv:1810.09519*, 2, 2018.

[26] A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.

[27] A. Langlois, D. Stehlé, and R. Steinfeld. Gghlite: More efficient multilinear maps from ideal lattices. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 239–256. Springer, 2014.

[28] M. Lecuyer, V. Atlidakis, R. Geambasu, D. Hsu, and S. Jana. Certified robustness to adversarial examples with differential privacy. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 656–672. IEEE, 2019.

[29] B. Li, C. Chen, W. Wang, and L. Carin. Certified adversarial robustness with additive noise. In *Advances in Neural Information Processing Systems*, pages 9464–9474, 2019.

[30] X. Liu, M. Cheng, H. Zhang, and C.-J. Hsieh. Towards robust neural networks via random self-ensemble. In *European Conference on Computer Vision*, pages 381–397. Springer, 2018.

[31] D. Lowd and C. Meek. Adversarial learning. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 641–647, 2005.

[32] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

[33] J. H. Metzen, T. Genewein, V. Fischer, and B. Bischoff. On detecting adversarial perturbations. In *Proceedings of 5th International Conference on Learning Representations (ICLR)*, 2017.

[34] S. Mitaim and B. Kosko. Adaptive stochastic resonance. *Proceedings of the IEEE*, 86(11):2152–2183, 1998.

[35] M. Mohri, A. Rostamizadeh, and A. Talwalkar. *Foundations of machine learning*. 2018.

[36] N. Papernot, P. McDaniel, X. Wu, S. Jha, and A. Swami. Distillation as a defense to adversarial perturbations against deep neural networks. In *2016 IEEE Symposium on Security and Privacy (SP)*, pages 582–597. IEEE, 2016.

[37] L. Perez and J. Wang. The effectiveness of data augmentation in image classification using deep learning. *arXiv preprint arXiv:1712.04621*, 2017.

[38] G. Peyré, M. Cuturi, et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019.

[39] P. Rafael, M. Laurent, A. Alexandre, K. Hisashi, Y. Florian, G.-P. Cédric, and A. Jamal. Theoretical evidence for adversarial robustness through randomization. In *Advances in Neural Information Processing Systems*, pages 11838–11848, 2019.

[40] A. S. Rakin, Z. He, and D. Fan. Parametric noise injection: Trainable randomness to improve deep neural network robustness against adversarial attack. *arXiv preprint arXiv:1811.09310*, 2018.

[41] A. Rényi. On measures of entropy and information. Technical report, Hungarian Academy of Sciences Budapest Hungary, 1961.

[42] C. Robert. *The Bayesian choice: from decision-theoretic foundations to computational implementation*. Springer Science & Business Media, 2007.

[43] H. Salman, J. Li, I. Razenshteyn, P. Zhang, H. Zhang, S. Bubeck, and G. Yang. Provably robust deep learning via adversarially trained smoothed classifiers. In *Advances in Neural Information Processing Systems*, pages 11289–11300, 2019.

[44] L. Schmidt, S. Santurkar, D. Tsipras, K. Talwar, and A. Madry. Adversarially robust generalization requires more data. In *Advances in Neural Information Processing Systems*, pages 5014–5026, 2018.

[45] S. Shalev-Shwartz and S. Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

[46] M. Sharif, S. Bhagavatula, L. Bauer, and M. K. Reiter. Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition. In *Proceedings of the 2016 acm sigsac conference on computer and communications security*, pages 1528–1540, 2016.

[47] C.-J. Simon-Gabriel, Y. Ollivier, L. Bottou, B. Schölkopf, and D. Lopez-Paz. First-order adversarial vulnerability of neural networks and input dimension. In *International Conference on Machine Learning*, pages 5809–5817, 2019.

[48] C. Sitawarin, A. N. Bhagoji, A. Mosenia, M. Chiang, and P. Mittal. Darts: Deceiving autonomous cars with toxic signs. *arXiv preprint arXiv:1802.06430*, 2018.

[49] D. Su, H. Zhang, H. Chen, J. Yi, P.-Y. Chen, and Y. Gao. Is robustness the cost of accuracy?–a comprehensive study on the robustness of 18 deep image classification models. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 631–648, 2018.

[50] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations*, 2014.

[51] F. Tramer, N. Carlini, W. Brendel, and A. Madry. On adaptive attacks to adversarial example defenses. *arXiv preprint arXiv:2002.08347*, 2020.

[52] D. Tsipras, S. Santurkar, L. Engstrom, A. Turner, and A. Madry. Robustness may be at odds with accuracy. *International Conference on Learning Representation*, 2019.

[53] I. Vajda. Note on discrimination information and variation. *IEEE Trans. Inform. Theory*, 16(6):771–773, 1970.

[54] A. W. Van der Vaart. *Asymptotic statistics*, volume 3. Cambridge university press, 2000.

[55] T. van Erven and P. Harremos. Rényi divergence and kullback-leibler divergence. *IEEE Transactions on Information Theory*, 60(7):3797–3820, 2014.

[56] G. Verma and A. Swami. Error correcting output codes improve probability estimation and adversarial robustness of deep neural networks. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d' Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8646–8656. Curran Associates, Inc., 2019.

[57] C. Villani. *Topics in optimal transportation*. Number 58. American Mathematical Soc., 2003.

[58] C. Xie, J. Wang, Z. Zhang, Z. Ren, and A. Yuille. Mitigating adversarial effects through randomization. In *International Conference on Learning Representations*, 2018.

[59] C. Xie, J. Wang, Z. Zhang, Z. Ren, and A. Yuille. Mitigating adversarial effects through randomization. In *International Conference on Learning Representations*, 2018.

[60] H. Xu and S. Mannor. Robustness and generalization. *Machine learning*, 86(3):391–423, 2012.

[61] D. Yao, Z. Xi, Z. Tianyi, C. Chen, L. Guannan, and K. Miryung. An analysis of adversarial attacks and defenses on autonomous driving models. In *18th Annual IEEE International Conference on Pervasive Computing and Communications*. IEEE, 2020.

[62] D. Yin, R. Kannan, and P. Bartlett. Rademacher complexity for adversarially robust generalization. In *International Conference on Machine Learning*, pages 7085–7094, 2019.

[63] S. Zagoruyko and N. Komodakis. Wide residual networks. In *Proceedings of the British Machine Vision Conference (BMVC)*, pages 87.1–87.12. BMVA Press, 2016.

[64] H. Zhang, Y. Yu, J. Jiao, E. P. Xing, L. E. Ghaoui, and M. I. Jordan. Theoretically principled trade-off between robustness and accuracy. *International conference on Machine Learning*, 2019.

[65] S. Zozor and P.-O. Amblard. Stochastic resonance in discrete time nonlinear AR(1) models. *IEEE transactions on Signal Processing*, 47(1):108–122, 1999.