Robustness Should Not Be at Odds with Accuracy

Sadia Chowdhury[1]
EECS Department, York University, Toronto, Canada

Ruth Urner
EECS Department, York University, Toronto, Canada

Abstract

The phenomenon of adversarial examples in deep learning models has caused substantial concern over their reliability and trustworthiness: in many instances an imperceptible perturbation can falsely flip a neural network's prediction. Applied research in this area has mostly focused on developing novel adversarial attack strategies or building better defenses against such. It has repeatedly been pointed out that adversarial robustness may be in conflict with requirements for high accuracy. In this work, we take a more principled look at modeling the phenomenon of adversarial examples. We argue that deciding whether a model's label change under a small perturbation is justified, should be done in compliance with the underlying data-generating process. Through a series of formal constructions, systematically analyzing the relation between standard Bayes classifiers and robust-Bayes classifiers, we make the case for adversarial robustness as a locally adaptive measure. We propose a novel way defining such a locally adaptive robust loss, show that it has a natural empirical counterpart, and develop resulting algorithmic guidance in form of data-informed adaptive robustness radius. We prove that our adaptive robust data-augmentation maintains consistency of 1-nearest neighbor classification under deterministic labels and thereby argue that robustness should not be at odds with accuracy.

2012 ACM Subject Classification Theory of computation → Machine learning theory

Keywords and phrases Statistical Learning Theory, Bayes optimal classifier, adversarial perturbations, adaptive robust loss

Digital Object Identifier 10.4230/LIPIcs.FORC.2022.5

Related Version *Previous Version*: https://arxiv.org/abs/2106.13326

Funding This research was supported by an NSERC discovery grant.

Acknowledgements Ruth Urner is also faculty affiliate member of Toronto's Vector Institute.

1 Introduction

Deep learning methods have enjoyed phenomenal successes on a wide range of applications of predictive tasks in the past decade. However, it has been demonstrated that, while these networks are often highly accurate at making predictions on natural data inputs, the performance can degrade drastically when inputs are slightly manipulated [32]. Flipping a few pixels in an image, a perturbation that is not perceivable by humans, can lead to misclassification by the trained network. These unexpected, and seemingly erratic behaviors of deep learning models have caused substantial concern over their reliability and trustworthiness. Particularly so, if these models are to be employed in applications where vulnerability to manipulations may have fatal consequences (for example if learning based vision technologies are to be employed in self-driving cars).

[1] This research was done while Sadia Chowdhury was a graduate student at York University, Toronto, Canada.

Recent years have thus seen a surge in studies aiming to enhance robustness of deep learning [10, 18, 1]. Practical approaches often either smooth out a trained predictor [12, 28], or augment the training data with perturbations of natural inputs as a way to promote robustness [36, 41]. In adversarial training this is done as part of the optimization [20, 24]. On the other hand, studies on the theory of adversarial robustness have often focused on exploring unexpected gaps in statistical and computational complexity when learning under an adversarial loss as opposed to the standard binary classification loss [26, 40, 25]. Numerous studies, both theoretical and practical, have pointed out that increasing robustness often comes at a cost of lower predictive accuracy [15, 19, 27, 5].

Naturally, an important component of analyzing and exploring a real world phenomenon, such as adversarial perturbations, is formalizing it appropriately. In supervised machine learning, the learning objective is typically encoded in form of a loss function. In this work, we take a principled look at the common definition of adversarial loss. Both theoretical studies and practical heuristics developed in the context of promoting robustness to adversarial attacks are mostly aimed at a fixed notion of smoothness with a fixed degree of perturbations that the model should be made robust to. In contrast, we formally argue how the notion of what is an admissible adversarial perturbation should be *informed by the data*. That is, robustness requirements should be aligned with the underlying data-generating process. We show how such an alignment inherently requires a *locally adaptive notion of robustness, that is, a locally adaptive robust loss*.

More specifically, we start by analyzing carefully how the previously established trade-offs between accuracy and robustness depend on a chosen (fixed) robustness parameter and the probability mass close to the decision boundary of the true underlying data-generating process. We introduce a new notion to quantify this trade-off, the *margin rate* of the distribution. We prove that, given the margin rate of a distribution, a robustness parameter can be chosen so that the two predictors that are optimal with respect to accuracy and optimal with respect to robustness loss respectively, have similar loss values (in terms of both classification and robust loss). However, we also show that choosing the robustness parameter slightly too large, can result in those two optimal predictors be very different functions. They may assign different labels on half of the space (with probability 0.5 over the data-generating process). This means that, if the robustness parameter is chosen even slightly too large, any learning method that *converges to the best possible robust loss as training data set size increases, may converge to a predictor with classification error 0.5*!

This motivates our proposition of redefining the robustness requirement. We argue that *robustness is inherently a local property* and that learned predictors should thus satisfy a local notion of robustness that is in line with the underlying data-generating process. While such a requirement can not readily be phrased as a loss function (that operates on a pair of predictor and input/output data instance), we derive a natural empirical version of this requirement. This allows for evaluating the novel adaptive robustness requirement on datasets. Further, we show how our notion of locally adaptive robustness yields a natural way of determining the robustness radius for data-augmentation. This could be used either for data-augmentation as a preprocessig step or for advesarial training.

Finally, we prove that using this form of data-augmentation as a pre-processing step maintains consistency of 1-nearest neighbor classification on tasks without stochasticity in the labels. That is, a nearest neighbor classifier on an adaptively augmented dataset converges to the optimal classification accuracy, while also satisfying the requirements of the adaptive robust loss. This formally shows how our novel framework resolves the conflicts with accuracy that are inherent in any non-adaptive notions of robustness.

1.1 Overview and summary of main contributions

We introduce our formal setup, notation for loss functions, optimal predictors and notions of statistical consistency in Section 3. In Section 4, we start with a few simple constructions, exploring how robustness (and potential divergence of 0/1-optimal and robust-optimal classifiers) relates to margins and separability of the underlying data-generating distribution. Our main contributions are presented in Sections 5 and 6 and can be summarized as follows:

Margin rate and margin canonical Bayes predictor. In Section 5.1, we *introduce the notion of a margin canonical Bayes predictor* and *the margin-rate* (Definition 4). The margin canonical Bayes predictor is a classifier that is optimal both in terms of accuracy and in terms of margins (in a precise sense that we define in this section). The margin rate can be viewed as a *relaxed measure of distributional class separateness*. It is relaxed in the sense that is does not enforce a hard margin between different classes (which is an unrealistic requirement) and instead even allows for overlap between the two class-conditional marginals (resulting in stochastic labels). We then relate the margin rate to suitable choices of r. We prove that, given the margin rate, we can choose the robustness parameter so that optimal predictors for the binary loss are also close to optimal with respect to the robust loss and vice versa (Theorem 5). Further, we show that if the labels are deterministic (no overlap between the two class-conditional marginals), then these are also close *as functions*. However, we also show that the non-stochasticity of the labels is necessary for the functions to be guaranteed to be close and that choosing r slightly too large can lead to large differences in the optimal predictors (Observations 6 and 7). Subsequently, in Subsection 5.2 we argue that, if the distribution has inherently different scales of robustness in different parts of the space, then even under deterministic labels choosing r suitable according to Theorem 5 does not lead to what is intuitively desired of a robust predictor.

Redefining robustness and resolving the conflicts with accuracy. The analysis outlined above leads to our proposition to *re-define robustness as a locally adaptive requirement*. This is presented in Section 6. There, we *introduce the adaptive robust loss*, define its *empirical version*, and develop guidance for adaptive robust data augmentation. Our proposed definition implies that the optimal predictors with respect to the binary loss and the adaptive robust loss coincide. Further, we prove that our adaptive robust data-augmentation *maintains consistency* of 1-nearest neighbor classification (NN) under deterministic labels. This shows that the undesirable effect of robustness being "at odds" with accuracy is an artifact of a specific, though common, way of defining robustness. It can be avoided be letting robustness requirements be informed by the underlying data-generating process.

Illustrative visualizations. Finally, in Appendix Section A we present a set of *illustrative experiments* for the proposed data-augmentation method and adaptive robust loss in combination with training a ReLU neural network. The synthetic datasets were designed so as to highlight the occurrence of adversarial examples when the data sits on a lower dimensional manifold, a scenario that is considered one of the sources adversarial vulnerability [22]. Our experiments visually make the case for the adaptive robust loss in situations where the label classes have *different degrees of separation in different parts of the space*.

A note on generalizations. For concreteness, we focus our presentation in this work on binary classification and work with the Euclidian metric. However, our definitions and result straightforwardly generalize to multi-class classification and to other metrics (with suitably chosen covering numbers replacing the Euclidian dimension in our result on consistency under adaptive robust data-augmentation).

2 Related Work

Enhancing robustness to adversarial attacks has received an enormous amount of research attention in recent years, in particular in terms of practical advancements [10, 18, 1, 9, 21]. We will focus our discussion of prior work on studies relating to theoretical aspects of learning under a robust loss.

Numerous recent theoretical studies focus on the parametric setup and analyze how introducing a robustness requirement may affect statistical convergence of the induced loss classes [13, 29, 26, 40, 2], whereas others have focused on computational implications [4, 25]. In particular, that there can be arbitrarily large gaps between the sample complexity of learning a hypothesis with respect to classification versus roust loss [13, 26]. Several studies have derived convergence bounds for classification under adversarial manipulations for fixed hypothesis classes [16, 3, 8].

Most related to our work are recent studies that also discuss possible options (and their implications) for phrasing a robust loss [15, 19], and in particular studies that pointed out and analyzes the trade-off between accuracy and robustness [17, 33, 39]. In particular, a recent study systematically explored the relationship between (a notion of local) Lipschitzness of a nearest neighbor predictor and its robustness. Further closely related to our work are recent studies that analyze and derive properties of optimal predictors under the robust loss and their relation to nearest neighbor predictors [35, 6, 38]. The latter work studies non-parametric learning for robust classification and proposes a method of data-preprocessing, and, similar to our result for 1-Nearest Neighbor prediction, proves implied consistency. However, the pre-processing in that study consists of pruning rather than augmenting the data. However, robustness in these prior works is considered with respect to a fixed robustness parameter. In this work, we carefully argue that adversarial robustness should instead be phrased as a locally adaptive requirement. Recently, a similar argument has independently been made [7]. Ideas of a locally adaptive robustness parameter have also appeared in some practical developments on refining adversarial training [5, 14]. Our work can be viewed as providing a formal foundation to those ideas, cleanly relating the concept of adaptive robustness to the distribution that models the data-generating process, as well as formally showing how a fixed robustness parameter easily yields inconsistencies between the robust and the standard classification loss.

Finally, we note that relationship between non-parametric methods and local adaptivity is well established and our work builds on this. In particular, it has been shown shown that nearest neighbor methods' convergence can be understood and quantified in terms of local smoothness properties of the underlying data-generating process for regression [23] as well as for classification tasks [11].

3 Formal Setup

3.1 Basic notions of statistical learning

We employ a standard setup of statistical learning theory for classification. We let $\mathcal{X} \subseteq \mathbb{R}^d$ denote the domain and $\mathcal{Y}$ (mostly $\mathcal{Y} = \{0, 1\}$) a (binary) label space. We assume that data is generated by some distribution P over $\mathcal{X} \times \mathcal{Y}$ and let $P_{\mathcal{X}}$ denote the marginal of P over $\mathcal{X}$. We let $\mathrm{supp}(P_{\mathcal{X}})$ denote the support of this marginal. Further, we use notation $\eta_P(x) = \mathbb{P}_{(x,y)\sim P}[y = 1 \mid x]$ to denote the *regression function* of P. We say that the distribution has *deterministic labels* if $\eta_P(x) \in \{0, 1\}$ for all $x \in \mathcal{X}$. A *classifier* or *hypothesis* is a function $h : \mathcal{X} \to \mathcal{Y}$. We let $\mathcal{F}$ denote the set of all Borel measurable functions from $\mathcal{X}$ to $\mathcal{Y}$ (or all functions in case of a countable domain). A *hypothesis class* is a subset of $\mathcal{F}$, often denoted by $\mathcal{H} \subseteq \mathcal{F}$.

The quality of prediction of a hypothesis on an input/output pair (x, y) is measured by a *loss function* $\ell : (\mathcal{F} \times \mathcal{X} \times \mathcal{Y}) \to \mathbb{R}$. For classification problems, the quality of prediction is typically measured with the *binary* or *classification loss*: $\ell^{0/1}(h, x, y) = \mathbb{1}[h(x) \neq y]$, where $\mathbb{1}[\alpha]$ denotes the indicator function for predicate α.

We denote the *expected loss* (or *true loss*) of a hypothesis h with respect to the distribution P and loss function ℓ by $\mathcal{L}_P(h) = \mathbb{E}_{(x,y)\sim P}[\ell(h, x, y)]$. In particular, we will denote the true binary loss by $\mathcal{L}_P^{0/1}(h)$. The *Bayes classifier* is a (in general not unique) classifier which has the minimal true loss with regard to P. We denote the Bayes classifier with respect to the binary loss as h_P^B and it's loss, the *Bayes risk* by $\mathcal{L}_P^B = \mathcal{L}_P^{0/1}(h_P^B)$

The *empirical loss* of a hypothesis h with respect to loss function ℓ and a sample $S = ((x_1, y_1), \ldots, (x_n, y_n))$ is defined as $\mathcal{L}_S(h) = \frac{1}{n}\sum_{i=1}^{n} \ell(h, x_i, y_i)$.

Further, we use the following notation to denote the set of domain points on which two classifiers differ: $h\Delta h' := \{x \in \mathcal{X} \mid h(x) \neq h'(x)\}$.

A *learner* $\mathcal{A}$ is a function that maps a finite sequence of labeled instances $S = ((x_1, y_1), \ldots$ $\ldots, (x_n, y_n))$ to a hypothesis $h = \mathcal{A}(S)$. The following notion of a *consistent learner* captures a basic desirable property: as the learner sees larger and larger samples from the data-generating distribution, the loss of the learner's output should converge to the Bayes risk.

▶ **Definition 1** (Consistency). *We say that a learner $\mathcal{A}$ is* consistent *with respect to a set of distributions $\mathcal{P}$ if, for every $P \in \mathcal{P}$, every $\epsilon, \delta > 0$ we have there is a sample-size $n(P, \epsilon, \delta)$ such that, for all $n \geq n(P, \epsilon, \delta)$, we have $\mathbb{P}_{S\sim P^n}\left[\mathcal{L}_P(\mathcal{A}(S)) \leq \mathcal{L}_P^B + \epsilon\right] \geq 1 - \delta$.*

3.2 (Adversarially) robust loss

We consider the most commonly used notion of an (adversarial) robust loss [26, 37]. For a point $x \in \mathcal{X}$, we let $\mathcal{B}_r(x)$ denote the (open) ball of radius r around x. We then define the robust loss as: $\ell^r(h, x, y) = \mathbb{1}[\exists z \in \mathcal{B}_r : h(z) \neq y]$ and we let $\mathcal{L}_P^r(h)$ denote the expected robust loss of h.

As has been done in prior work, we decompose the robust loss into its error and margin components [42, 2]: We have $\ell^r(h, x, y) = 1$ if and only if h makes a mistake on x with respect to label y, or, there is an r-close instance $z \in \mathcal{B}_r(x)$ that h labels different than x, that is, x is r-close to h's decision boundary.

The first condition holds when (x, y) falls into the *error region*, $\mathrm{err}[h] = \{(x, y) \in X \times Y) \mid h(x) \neq y\}$. The second condition holds when x lies in the *margin area* of h. We define the *margin area* of h, as the subset $\mathrm{mar}[h, r] \subseteq X$ defined by

$$\mathrm{mar}[h, r] = \{x \in \mathcal{X} \mid \exists z \in \mathcal{B}_r(x) : h(x) \neq h(z)\}$$

We can define notions of a Bayes classifier, and consistency of a learner $\mathcal{A}$ with respect to the robust loss analogously to these notions for the binary loss. We will denote the robust-Bayes classifier by h_P^{rB} and the robust-Bayes risk by $\mathcal{L}_P^{rB} = \mathcal{L}_P^r(h_P^{rB})$. We will often simply refer to the Bayes predictors as the 0/1-optimal or the r-robust optimal predictors. We note that these optimal predictors are not unique, in particular in the case that the support of the marginal $P_{\mathcal{X}}$ does not cover the full space. For example, if the data-generating distribution is supported on a lower dimensional manifold, then a 0/1-optimal predictor is only uniquely determined on that manifold (and even there only with exception of 0-mass subsets and not in areas with $\eta_P(x) = 0.5$). Similarly, r-robust optimality can be fulfilled by various predictors if the data-generating distribution is strongly separable (see Definition 4). Explicit forms (analogous to the 0/1-Bayes being a threshold of the regression function) of the r-robust optimal predictor have been derived in the literature ([38]).

4 Robustness and Margins

In this section, as a warm-up, we investigate implications of the existence of a low robust-loss classifier and differences between low binary and low robust loss. We show that the optimal classifiers with respect to these losses can differ significantly, implying that optimizing for one can strongly hurt performance with respect to the other. We then analyze the relationship between the existence of robust classifiers and margin (or separability) properties of the underlying data-generating process. We argue that, while separability implies the existence of robust classifiers with respect to *some* robustness parameter r, using a fixed robustness parameter can contravene the intention of deriving predictors that are both accurate and as robust as possible.

4.1 Binary optimal versus robust optimal

It has been shown before that the definition of the r-robust loss implies that, even in situations where the 0/1-Bayes risk is 0, that is where the labels are deterministic, no classifier may have 0 robust loss [15, 33, 42, 19]: The existence of a classifier h with $\mathcal{L}^r_P(h) = 0$ implies that the distribution is *separable*, that is, $P_{\mathcal{X}}$ is supported on r-separated regions of $\mathcal{X}$ and these regions are label-homogeneous. Namely, $\mathcal{L}^r_P(h) = 0$ implies $\mathcal{L}^{0/1}_P(h) = 0$, which means that the labeling of P is deterministic. In addition, we must have $P(\mathrm{mar}[h, r]) = 0$, which implies that any point x in the support of $P_{\mathcal{X}}$ with $h(x) = 1$ has distance at least $2r$ from any point in that support with $h(x) = 0$. In this case, this function $h = h^B_P = h^{rB}_P$ is optimal with respect to both losses.

In this subsection we inspect the potential tension between robustness and accuracy with an emphasis on the role that stochasticity of the labels play in this phenomenon. We start by observing that even if the labels are not necessarily deterministic, the optimal robust loss is strictly larger than the optimal 0/1-loss if and only if a Bayes classifier does not have a strict margin.

▶ **Theorem 2.** *We have $\mathcal{L}^{rB}_P = \mathcal{L}^B_P$ if and only if there exists a 0/1-optimal classifier h^B_P with $P_{\mathcal{X}}(\mathrm{mar}[h^B_P, r]) = 0$.*

Proof. We first assume that $P_{\mathcal{X}}(\mathrm{mar}[h, r]) > 0$ for all classifiers h that are 0/1-optimal. We fix one of them and denote it by h^B_P. Then $\mathcal{L}^r_P(h^B_P) > \mathcal{L}_P(h^B_P) = \mathcal{L}^B_P$, since on every point in its margin area, h^B_P suffers binary loss at most 0.5, while it suffers robust loss 1. Outside the margin area the loss contributions are identical for both loss functions. Furthermore, for any classifier h that is not 0/1-optimal, we have $\mathcal{L}^r_P(h) \geq \mathcal{L}^{0/1}_P(h) > \mathcal{L}^B_P$. Thus, independently of whether an optimal robust classifier h^{rB}_P is also 0/1-optimal or not, we have $\mathcal{L}^{rB}_P = \mathcal{L}^r_P(h^{rB}_P) > \mathcal{L}^B_P$.

As for the other direction, if there is a 0/1-optimal classifier h^B_P with $P_{\mathcal{X}}(\mathrm{mar}[h^B_P, r]) = 0$, then it follows immediately, that this classifier is also optimal with respect to the robust loss and its robust loss is identical to its binary loss. Thus $\mathcal{L}^{rB}_P = \mathcal{L}^B_P$. ◀

Moreover, we will now see, that if the data-generating distribution does not have a margin in the above strong sense, then the optimal classifiers with respect to 0/1-loss and r-robust loss can differ significantly *as functions*. The construction for the below result has (in very similar form) appeared in earlier work [42].

▶ **Theorem 3.** *Let $r > 0$ be a robustness parameter. There exist distributions P such that any predictors h^B_P and h^{rB}_P that are optimal with respect to 0/1-loss and r-robust loss respectively, satisfy $P_{\mathcal{X}}[h^B_P \,\Delta\, h^{rB}_P] = \frac{1}{2}$, where $h^B_P \,\Delta\, h^{rB}_P = \{x \in \mathcal{X} \mid h^B_P(x) \neq h^{rB}_P(x)\}$ is the set of domain points on which the two optimal classifiers differ.*

Proof. We consider a distribution P, where $P_{\mathcal{X}}$ is supported (uniformly) on just two points x_0 and x_1 at distance less than r from each other. x_0 is always generated with label 0 and x_1 is always generated with label 1. Clearly, the 0/1-optimal classifier h_P^B labels accordingly: $h_P^B(x_0) = 0$ and $h_P^B(x_1) = 1$, resulting in $\mathcal{L}_P^{0/1}(h_P^B) = 0$. However, this classifier has largest possible r-robust loss: $\mathcal{L}_P^r(h_P^B) = 1$, since both points are at distance less than r from a point that h_P^B labels differently. On the other hand, any constant function h_c has robust loss $\mathcal{L}_P^r(h_c) = 1/2$, since it's margin has weight 0 and it mislabels with probability $1/2$. This is optimal with respect to the r-robust loss. Thus, we showed that $P_{\mathcal{X}}[h_P^B \,\Delta\, h_P^{rB}] = \frac{1}{2}$. ◀

The example in the above proof shows that binary and robust optimal predictors can differ in half the area of the space. In particular, when the robustness radius r is not chosen suitably, optimizing for one can be strongly sub-optimal (incurring regret of $1/2$) for the other. This means that *any learning method, will be inconsistent with respect to at least one of the two losses in question.*

Of course, in the above example, the robustness parameter and distribution are constructed to not match suitably.

5 Relaxations of separability and the margin canonical Bayes

Strict separability between the label classes, as considered in the previous section, is a very strong assumption. We extend and refine the arguments in the previous section by relaxing this requirement and showing that, one can choose the robustness parameter r in dependence on "how separable" (in a precise sense that we introduce next) the distribution P is and on how close we would like the optimal predictors to be.

5.1 Choosing a robustness parameter

Note that, for a fixed predictor h, we have $P_{\mathcal{X}}(\mathrm{mar}[h, r]) \geq P_{\mathcal{X}}(\mathrm{mar}[h, r'])$ if $r \geq r'$. Thus, we can define a function

$$\phi_P^h(r) = P_{\mathcal{X}}(\mathrm{mar}[h, r])$$

which will monotonically decrease to 0 as r goes to 0 for any predictor h. If h is a Bayes predictor, then the rate at which $\phi_P^h(r)$ converges to 0 as $r \to 0$, can be viewed as a measure of "how separable" the data- generating process is, that is, how fast the density of the marginal $P_{\mathcal{X}}$ vanishes towards the boundary between the two label classes. However, since the Bayes predictor is generally not uniquely defined, we need to specify which Bayes predictor should be employed to serve as a measure of the separability of the distribution. For simplicity, we will assume here that we have $\eta_P(x) \neq 0.5$ for the regression function with probability 1. Then we define a margin-canonical Bayes predictor as follows: We let $\mathcal{X}^0 \subseteq \mathrm{supp}(P_{\mathcal{X}})$ denote the closure of the part of the space, where $\eta_P(x) < 0.5$ and let $\mathcal{X}^1 \subseteq \mathrm{supp}(P_{\mathcal{X}})$ the closure of the part of the space where $\eta_P(x) > 0.5$. That is, under the above assumption, the support of the marginal $P_{\mathcal{X}}$ is $\mathcal{X}^0 \cup \mathcal{X}^1$.

We can now define a *margin-canonical Bayes classifier* h_P^B by nearest neighbor labeling with respect to the sets $\mathcal{X}^0$ and $\mathcal{X}^1$. We only need to specify $h_P^B(x)$ for points x that are outside the support of $P_{\mathcal{X}}$. By definition, there exists a ball of some radius r around such a point x that has has no probability mass: $P_{\mathcal{X}}(\mathcal{B}_r(x)) = 0$. Thus, x has positive distance to both $\mathcal{X}^0$ and $\mathcal{X}^1$ and we will set $h_P^B(x) = i$ if $\mathcal{X}^i$ is the closer set to x, breaking ties arbitrarily. We note that our definitions and results in subsequent sections also hold for the margin rate of any other Bayes classifier.

▶ **Definition 4** (Margin rate). *Let P be a distribution over $\mathcal{X} \times \{0,1\}$ and let h_P^B be the margin-canonical Bayes classifier. Then we define* margin-rate *of P as the function $\Phi_P(r) = \phi_P^{h_P^B}(r)$. If there exists an $r > 0$ such that $\Phi_P(r) = 0$, we call the distribution P* strongly separable.

The margin rate is related the notion of *Probabilistic Lipschitzness* [34] and the *geometric noise exponent* [31]. We now show how the margin rate can be used for the suitable choice of robustness parameter r. We show below how to choose a robustness parameter for which the optimal robust predictor has close to optimal classification loss and vice versa. If the labels of the distribution are deterministic, then we also get closeness as functions of the optimal predictors.

▶ **Theorem 5.** *Let P be a data-generating distribution over $\mathcal{X} \times \{0,1\}$, let $\Phi_P : \mathbb{R}^+ \to [0,1]$ denote its margin rate, and let h_P^B denote the 0/1-optimal classifier defining the margin rate. For every $\epsilon > 0$, if we let $r \in \Phi_P^{-1}([0,\epsilon])$, then for any r-robust optimal classifier h_P^{rB} we have $\mathcal{L}_P^r(h_P^B) \leq \mathcal{L}_P^{rB} + \epsilon$ and $\mathcal{L}_P^{0/1}(h_P^{rB}) \leq \mathcal{L}_P^B + \epsilon$.*
In addition, if the labeling of P is deterministic, we have $P_{\mathcal{X}}[h_P^B \,\Delta\, h_P^{rB}] \leq \epsilon$.

Proof of Theorem 5. Due to the way we chose the robustness parameter r here, we immediately get

$$\mathcal{L}_P^r(h_P^B) \leq \mathcal{L}_P^{0/1}(h_P^B) + \epsilon = \mathcal{L}_P^B + \epsilon$$

since $P(\mathrm{mar}[h_P^B, r]) \leq \epsilon$. We need to argue, that no other classifier h can have significantly smaller robust loss. As in the proof of Theorem 2, we observe that, we have $\mathcal{L}_P^r(h) \geq \mathcal{L}_P^{0/1}(h) \geq \mathcal{L}_P^B$ for any classifier h. Thus, in particular $\mathcal{L}_P^r(h_P^{rB}) = \mathcal{L}_P^{rB} \geq \mathcal{L}_P^B$, which yields the first claim.

For the second inequality observe that h_P^B has r-robust loss at most $\mathcal{L}_P^B + \epsilon$ by choice of r. Any robust-optimal classifier h_P^{rB} therefore has robust loss at most $\mathcal{L}_P^B + \epsilon$, which implies that its binary loss is bounded by the same quantity.

Now we assume that the labeling of P is deterministic. This implies that $\mathcal{L}_P^{0/1}(h_P^B) = 0$, thus $\mathcal{L}_P^r(h_P^B) = \mathcal{P}_{\mathcal{X}}(\mathrm{mar}[h_P^B, r])$. Let h_P^{rB} be a robust-optimal classifier. By definition of being robust-optimal, we have $\mathcal{L}_P^r(h_P^{rB}) \leq \mathcal{L}_P^r(h_P^B) = \mathcal{P}_{\mathcal{X}}(\mathrm{mar}[h_P^B, r]) \leq \epsilon$. Thus, in particular $\mathcal{L}_P^{0/1}(h_P^{rB}) \leq \epsilon$, which, in the case of deterministic labels implies $P_{\mathcal{X}}[h_P^B \,\Delta\, h_P^{rB}] \leq \epsilon$. ◀

We next argue that, while a separability assumption can yield closeness in loss values of the optimal predictors, it implies closeness of the actual functions only if the labeling is also deterministic. That is, the assumption of deterministic labels is *necessary* for the second part of the above Theorem (Observation 6). More specifically, the result in the observation below shows that, a non-adaptive robustness parameter that will guarantee closeness of functions as in the first part of the above theorem, can not be determined as a function of the marginal distribution, but depends on a combination of the marginal and the "noise rate".

▶ **Observation 6.** *Let $\epsilon > 0$ be given. Then, for any γ with $0 < \gamma < \epsilon$, there exists a data-generating distribution P over $\mathbb{R}^2 \times \{0,1\}$ with linear margin rate $\Phi_P : \mathbb{R}^+ \to [0,1]$, $\Phi_P(r) = \min\{r, 1\}$ such that, for any $r \in \Phi_P^{-1}((\gamma, \epsilon))$, we get $P_{\mathcal{X}}[h_P^B \,\Delta\, h_P^{rB}] = \frac{1}{2}$*

Proof. We consider a uniform marginal over two rectangles in $\mathbb{R}^2$: We set $R_1 = [-2,-1] \times [-1,1]$ and $R_2 = [1,2] \times [-1,1]$. Further, we set the regression function

$$\eta(x_1, x_2) = \begin{cases} \frac{1}{2} + \gamma \text{ if } x_2 \geq 0 \\ \frac{1}{2} - \gamma \text{ if } x_2 \leq 0. \end{cases}$$

Now it follows that a 0/1-optima predictor is $h_P^B = \mathbb{1}[x_2 \geq 0]$ while, for any $r > \gamma$, we have $h_P^{rB} = \mathbb{1}[x_1 \geq 0]$, thus $P_{\mathcal{X}}[h_P^B \,\Delta\, h_P^{rB}] = \frac{1}{2}$. ◀

Next, we argue that, even under deterministic labels, choosing a robustness parameter slightly larger than implied by Theorem 5, can yield largely differing optimal predictors. The same construction as in the proof of Theorem 3 shows the following statement:

▶ **Observation 7.** *Let $\epsilon > 0$ be given. There exists a distribution P over $\mathbb{R} \times \{0,1\}$ that is strongly separable, such that, for any $r > \sup \Phi_P^{-1}([0,\epsilon])$, we have $P_{\mathcal{X}}[h_P^B \,\Delta\, h_P^{rB}] = \frac{1}{2}$.*

5.2 Towards local robustness

We now argue that, even if the distribution is strongly separable and the labels are deterministic, then choosing a uniform robustness parameter may not result in the desired outcomes in the following sense (see Figure 5.2): a classifier may be optimal with respect to the largest possible *fixed robustness parameter* (the orange classifier), but have a decision boundary that is unnecessarily close in some parts of the space where a larger *local robustness* would have been possible. To argue more formally, we consider a distribution over domain $\mathbb{R}^2 \times \{0,1\}$, where the support is distributed uniformly on four points, $((-1,0.9),0),((-1,1.1),1),((1,0.9),0),((1,2),1)$. Then predictor $h(x_1,x_2) = \mathbb{1}[x_2 \geq 1]$ is 0/1-optimal and also r-robust optimal for any $r \leq 0.1$. However, we may prefer a predictor h^* that keeps a larger distance from the point $(1,.9)$, and is equally optimal with respect to the 0.1-robust loss.

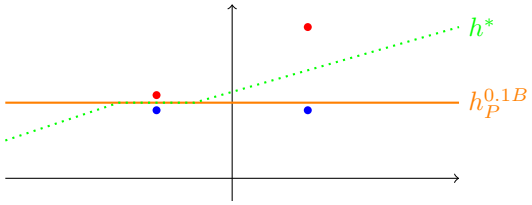

Figure 1 A robustness requirement with a uniform robustness radius is unsuitable here.

6 Redefining the Robustness Requirement

We have argued (Sections 4.1 and 5.1) that using a fixed robustness parameter r can lead to inconsistencies (in the sense that the optimal predictors with respect to binary and robust loss differ vastly) and that even under conditions where the optimal predictors can coincide (strong separability or suitably chosen robustness parameter), optimizing for the robust loss can lead to classifiers that do not reflect our intuition about an optimally robust predictor (Section 5.2). Ideally we would like a learned predictor to be *everywhere as robust as possible* (in the sense of the illustration in Figure 5.2). We will next formalize this intuition using the notions of the margin canonical Bayes and the margin rate, that we developed in the previous section.

6.1 A local robustness objective

We propose to phrase robustness in relation to a margin-canonical Bayes predictor. The core idea behind our definition is the following: If a margin-canonical Bayes predictor assigns a constant label in a ball $\mathcal{B}_r(x)$ around point x, then a robust predictor h should do the same (and only then!). For a predictor h and $x \in \mathcal{X}$, we let $\mathcal{B}^h(x)$ denote the largest ball around x on which h assigns a constant label (possibly $\mathcal{B}^h(x) = \{x\}$).

▶ **Definition 8** (Adaptive robustness). *Let P be a data-generating distribution h_P^B denote a margin-canonical Bayes predictor, and h an arbitrary predictor. We define the* adaptive robust loss ℓ^{ar} *as*

$$\ell^{ar}(h,x,y) = \mathbb{1}\left[h(x) \neq y \ \vee \ \mathcal{B}^{h_P^B}(x) \not\subseteq \mathcal{B}^h(x)\right].$$

That is, h suffers adaptive robust loss on point (x,y) if it misclassifies the point or if the point is closer to the decision boundary of h than to the decision boundary of the margin-canonical Bayes h_P^B. This definition implies that h_P^B has both minimal binary loss and optimal robust loss. We note that the above proposed loss is not technically a valid loss function, since it depends on h_P^B rather than just on h, x and y. Thus, we next propose a substitute notion of empirical adaptive robust loss.

6.2 Empirical adaptive robust loss

Let $S = ((x_1,y_1),\ldots,(x_n,y_n))$ be a labeled dataset. For a labeled domain point (x,y) we let $\rho_S(x)$ denote the distance from x to its nearest neighbor with opposite (or different in the case of more than two classes) label in S:

$$\rho_S(x,y) = \min_{i\in[n]}\{\|x_i - x\| \ \mid \ (x_i,y_i) \in S, y_i \neq y\}.$$

In the (degenerate) case that no such point in S has a label different from y (that is, all points in S have the same label), we set $\rho_S(x,y)$ to ∞ (or the diameter of the space). Note that $\rho_S(x,y)$ is well defined for points $(x,y) = (x_i,y_i) \in S$ from the dataset S itself. We now expand the dataset S by replacing each point with a (constant labeled) ball of radius $c \cdot \rho_S(x_i,y_i)$, for some (to be chosen) constant c.

▶ **Definition 9** (c-Adaptive robust expansion). *Let $S = ((x_1,y_1),\ldots,(x_n,y_n))$. We call the collection $S^c = ((\mathcal{B}_{c\cdot\rho_S(x_1,y_1)}(x_1),y_1),\ldots,(\mathcal{B}_{c\cdot\rho_S(x_n,y_n)}(x_n),y_n))$ the* c-adaptive robust expansion of S.

It is easy to see that, as long as $c \leq 1/2$, balls in the c-adaptive robust expansion of S overlap only if they have the same label. Thus, this expansion does not introduce any inconsistencies in the label requirements. Depending on the geometry of the data-generating process (eg. the curvature of the decision boundary of the regression function) we may also employ larger expansion parameters without introducing inconsistencies. Using the *c-adaptive robust expansion of S*, we can define an empirical version of the adaptive robust risk for fixed parameter c. For this, for a predictor $h : \mathcal{X} \to \mathcal{Y}$ and label y, we let $h^{-1}(y) \subseteq \mathcal{X}$ denote the part of the domain that h labels with y.

▶ **Definition 10** (Empirical c-adaptive robust loss). *Let c be an expansion parameter, $S = ((x_1,y_1),\ldots,(x_n,y_n))$ and $h : \mathcal{X} \to \mathcal{Y}$. We define the* empirical c-adaptive robust loss of h on S *as*

$$\mathcal{L}_S^{c-ar}(h) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}\left[\mathcal{B}_{c\cdot\rho_S(x_i,y_i)}(x_i) \not\subseteq h^{-1}(y_i)\right].$$

That is, a point $(x_i,y_i) \in S$ is counted towards the empirical c-adaptive robust empirical risk, if h does not label the whole ball $\mathcal{B}_{c\cdot\rho_S(x_i,y_i)}(x_i)$ in the expanded set with y_i.

Remark on connection between the population and empirical version

As is usual for an empirical loss, the empirical adaptive robust loss as defined above for $c = 0.5$ corresponds to the adaptive robust loss on the empirical distribution (that is a uniform distribution of the finite data sample).

6.3 Adaptive robust data-augmentation

While the empirical c-adaptive robust risk is well defined for any predictor h and dataset S, it may, computationally, not be straightforward to verify the condition $\mathbb{1}\left[\mathcal{B}_{c\cdot\rho_S(x,y)}(x,y) \nsubseteq h^{-1}(y)\right]$. A natural estimate is to use m uniform sample points $z^1,\ldots,z^m$ from the ball $\mathcal{B}_{c\cdot\rho_S(x,y)}(x)$ and verify whether h labels all of these with y. Similarly, for training purposes, we may want to use a sample version of the c-adaptive robust expansion of S. We call this the *m-sample-c-adaptive robust augmentation of S*. The so augmented dataset S^{mc} is a set of labeled domain points and can be used as a training data-set for a standard learning algorithm.

▶ **Definition 11** (Adaptive robust data augmentation)**.** *Let $S = ((x_1,y_1),\ldots,(x_n,y_n))$ be a labeled dataset, and $m \in \mathbb{N}$. We call the collection $S^{mc} = ((z_1^1,y_1),\ldots,(z_1^m,y_1),\ldots(z_n^1,y_n),\ldots,(z_n^m,y_n))$, where every z_i^j is uniformly sampled from the ball $\mathcal{B}_{c\cdot\rho_S(x_i,y_i)}(x_i)$, the* m-sample-c-adaptive robust augmentation of S.

To visualize the adaptive robust augmentation and its effects, we generated data from a "lower-dimensional manifold" in two dimensions, see Figure 2. It has been conjectured that the data being supported on a lower-dimensional manifold is a source of the phenomenon of vulnerability to small perturbations [22], which our visualization illustrates. The original support (the data-manifold) of data generating distributions can be seen as the green and blue lines in the first column of Figure 2, blue and green points representing points from the two classes. We trained a ReLU Neural Network with 2-hidden layers (of 10 neurons each) data points drawn from these shapes. The labeling behavior of the trained network is visualized over the ambient space in red and purple. The first column depicts the original, labeled data sets together with the networks trained on the original data. The next columns show the effect of augmentation and training with a fixed robustness parameter while the last column shows the adaptive robust augmentation.

The sequence of trained network illustrates how without augmentation, the network's decision boundary passes close to the data-manifold in several areas, yielding areas of adversarial vulnerability. The augmentation with fixed robustness, does not change this for small robustness radius. For larger, fixed robustness radius, the augmentation leads to blurring the labels. The last column shows how the adaptive robust augmentation changes the decision boundary of the trained network in the ambient space to "curve away" from the lower dimensional data manifold. Importantly the prediction on the data manifold remains unchanged. Thus the adaptive robust augmentation yields robustness without negatively affecting the accuracy of the predictor on the data-generating distribution.

We conjecture that most learners, that are consistent with respect to binary loss, remain consistent when fed a c-adaptive robust augmentation of S for $c \leq 1/2$. We prove this for a 1-nearest neighbor classification under deterministic labels. This result serves as evidence that our adaptive data augmentation does not induce any inconsistencies with the accuracy requirements. It holds for a c-robust augmentation and any m-sample-c-robust augmentation if $c \leq 0.5$. The proof has been moved to Appendix C.

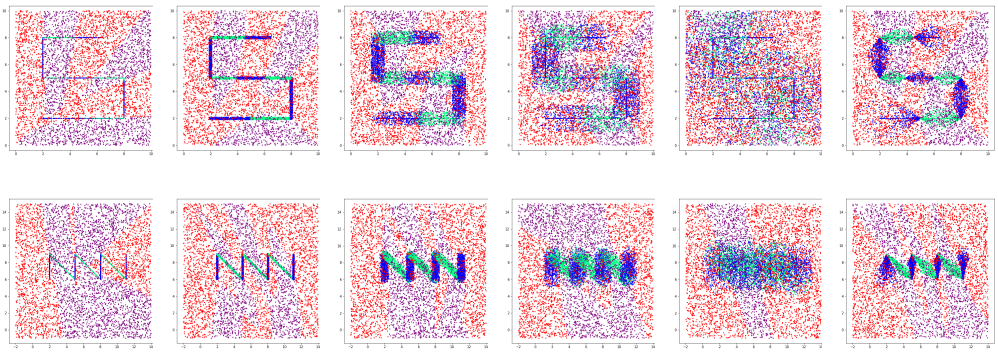

Figure 2 ReLU networks trained on data from a one-dimensional manifold, labeled with two classes (blue and green here). Left to right: original data, incrasing fixed augmentation parameters, and adaptive robust robust augmentation.

▶ **Theorem 12.** *Let P be a distribution over $[0,1]^d \times \{0,1\}$ with deterministic labels and margin rate $\Phi_P(r)$. Let $\epsilon, \delta > 0$ be given. Then, with probability at least $1-\delta$ over an is an i.i.d. sample S of size $n \geq \frac{3^d d^{0.5d}}{e\Phi_P^{-1}(\epsilon)^d \epsilon \delta}$ from P, the a 1-nearest neighbor predictor $h^{0.5}_{\mathrm{NN}}$ on a m-sample-0.5-adaptive robust augmentation of S satisfies $\mathcal{L}_P^{0/1}(h^{0.5}_{\mathrm{NN}}) \leq \epsilon$ for any $m \geq 1$.*

We will employ a similar proof technique as in Chapter 19 of [30]. In particular, we will employ Lemma 19.2 therein:

▶ **Lemma 13** (Lemma 19.2 in [30])**.** *Let $C_1, C_2, \ldots C_t$ be a collection of subsets of some domain set $\mathcal{X}$. Let D be a distribution over $\mathcal{X}$ and S be an iid sample from P of size n. Then $\mathbb{E}_{S\sim D^n}\left[\sum_{i:C_i\cap S=\emptyset} D(C_i)\right] \leq \frac{t}{n\cdot e}$.*

Recall that, for a labeled sample S, the collection $S^c = (\mathcal{B}_{c\cdot\rho_S(x_1,y_1)}(x_1,y_1),\ldots,\mathcal{B}_{c\cdot\rho_S(x_n,y_n)}(x_n,y_n))$ denotes the *c-adaptive robust expansion of S*. We will prove the theorem using this expansion for $c = 0.5$, but note, that the proof (and thus the Theorem) holds equally for

$$S^{mc} = ((z_1^1, y_1), \ldots, (z_1^m, y_1), \ldots (z_n^1, y_n), \ldots, (z_n^m, y_n)),$$

any *m-sample-c-adaptive robust augmentation of S* (where every z_i^j is uniformly sampled from the ball $\mathcal{B}_{c\cdot\rho_S(x_i,y_i)}(x_i)$).

Proof of Theorem 12. Let P be a distribution over $[0,1]^d \times \{0,1\}$ with deterministic labels and margin rate $\Phi_P(\cdot)$. We let h_P^B be a margin optimal Bayes predictor for P. Note that, since the labels of P are deterministic $\mathcal{L}_P^{0/1}(h_P^B) = 0$. Further, we let ϵ and δ be given and set $r = \Phi_P^{-1}(\epsilon)$ (to mean the largest r, such that $\Phi_P(r) \leq \epsilon$). Further, we set $r' = r/3$.

We can now partition the space $[0,1]^d$ into $t = \left(\frac{\sqrt{d}}{r'}\right)^d$ many sub-cubes of side-length $r'/\sqrt{d}$ and thus diameter r'. We denote the cells in this partition by $C_1, \ldots, C_t$.

We now let S be a labeled sample and let $h_S^c = h_S^5$ be the nearest neighbor classifier on the .5-adaptive robust expansion of S. We now bound the mass of points x on which h_S^c makes a false classification by noting that $h_S^c(x) \neq h_P^B(x)$ implies that one of these three conditions hold:

C1: x falls into a cell C_k that has empty intersection with the sample S;
C2: there is at least one sample point $(x_i, y_i) \in S$ in the same cell C_k as x, and there exists at least one such $(x_i, y_i) \in S$ with $y_i \neq h_P^B(x)$;
C3: there is at least one sample point $(x_i, y_i) \in S$ in the same cell C_k as x, and we have $y_i = h_P^B(x)$ for all (x_i, y_i) in the same cell, but there is another sample point $(x_j, y_j) \in S$ (in a different cell) with $y_j \neq h_P^B(x)$ and x is closer to the expansion $\mathcal{B}_{c\cdot\rho_S(x_j,y_j)}(x_j, y_j)$ of x_j than to the expansion $\mathcal{B}_{c\cdot\rho_S(x_i,y_i)}(x_i, y_i)$ for all x_i in C_k.

If S is an iid sample from P, then, by Lemma 13 the expected mass of points x cells that are not hit by the sample S is bounded by $\frac{t}{n\cdot \mathrm{e}} = \frac{3^d\sqrt{d}^d}{\Phi_P^{-1}(\epsilon)^d\cdot n\cdot \mathrm{e}}$. By Markov's inequality, this implies

$$\mathbb{P}_{S\sim P^n}\left[\sum_{i:C_i\cap S=\emptyset} P_{\mathcal{X}}(C_i) > \epsilon\right] \leq \frac{3^d\sqrt{d}^d}{\epsilon\cdot\Phi_P^{-1}(\epsilon)^d\cdot n\cdot \mathrm{e}}.$$

Setting this to δ shows that, with probability at least $1-\delta$ over a sample S of size $n \geq \frac{3^d\sqrt{d}^d}{\epsilon\cdot\Phi_P^{-1}(\epsilon)^d\cdot\delta\cdot \mathrm{e}}$ the mass of points that fall into "error case" C1 is bounded by ϵ.

We now argue that the mass of points that fall into "error case" C2 or C3 is also bounded by ϵ by showing that such points actually fall into the r-margin area of h_P^B and, by choice of r and by definition of Φ_P, we have $P_{\mathcal{X}}(\mathrm{mar}[r, h_P^B]) \leq \epsilon$.

Consider a point x in case C2. If there exist a point $(x_i, y_i) \in S$ in the same cell as x with $y_i \neq h_P^B(x)$, then by the choice of the size of the cells $x \in \mathrm{mar}[r', h_P^B] \subseteq \mathrm{mar}[r, h_P^B]$.

Now consider a point x in case C3: There exists at least one point $(x_i, y_i) \in S$ in the same cell as x and all points in the same cell as x have label $h_P^B(x)$. But there is another sample point $(x_j, y_j) \in S$ (in a different cell) with $y_j \neq h_P^B(x)$ and x is closer to the expansion $\mathcal{B}_{c\cdot\rho_S(x_j,y_j)}(x_j, y_j)$ of x_j than to the expansion $\mathcal{B}_{c\cdot\rho_S(x_i,y_i)}(x_i, y_i)$ for any x_i in the same cell as x, where $c = 0.5$.

Recall that $\rho_S(x_j, y_j)$ is the distance between x_j and a closest point in S of opposite label to y_j. We now set $\rho = 0.5\cdot\rho_S(x_j, y_j)$ for short, that is ρ is the radius of the expansion of (x_j, y_j). Since the cell that x is in also contains (x_i, y_i) and $y_i \neq y_j$ in this case C3, we know that

$$2\rho \leq \|x_i - x_j\|. \tag{1}$$

Further, we know

$$\|x_i - x\| \leq r' \tag{2}$$

since x_i in in the same cell as x.

Let $z \in \mathcal{B}_{c\cdot\rho_S(x_j,y_j)}(x_j, y_j)$ be the point in $\mathcal{B}_{c\cdot\rho_S(x_j,y_j)}(x_j, y_j)$ closest to x. Note that, since z in in the expansion of x_j, we have

$$\|z - x_j\| \leq \rho. \tag{3}$$

Then, since x is closer to the expansion of x_j than the expansion of x_i, we can infer, using Equation 2 that

$$\|x - z\| \leq \|x - x_i\| \leq r' = r/3. \tag{4}$$

This implies using the triangle inequality and Equations 4 and 2 that

$$\|z - x_i\| \leq \|z - x\| + \|x - x_i\| \leq 2r'. \tag{5}$$

Now, by again using the triangle inequality and Equations 3 we get

$$\|x_i - x_j\| \leq \|x_i - z\| + \|z - x_j\| \leq \|x_i - z\| + \rho, \tag{6}$$

Thus, using Equations 1 and then Equation 6, we get

$$2\rho \leq \|x_i - x_j\| \leq \|x_i - z\| + \rho$$

which immediately implies $\rho \leq \|x_i - z\|$. Together with Equation 5 the above yields: $\rho \leq 2r'$. Now, again invoking the triangle inequality and using Equations 4 and 3, we can bound the distance between x and x_j:

$$\|x - x_j\| \leq \|x - z\| + \|z - x_j\| \leq r' + 2r' = r.$$

Thus, in this case, x also falls into the r-margin area of h_P^B since $h_P^B(x) \neq h_P^B(x_j)$. This concludes the proof of the Theorem. ◀

7 Concluding Remarks

In this work, we provide a formal foundation for adversarial robustness as an *adaptive requirement*. We argue for re-framing adversarial robustness as a requirement that should be in line with the underlying distribution's margin properties. We do this by introducing a novel notion of the *margin-rate* that quantifies probability mass in proximity to a Bayes optimal's decision boundary in a more flexible way than standard notions of margin-separability do. We employ this measure to propose a formal notion of such an adaptive loss, as well as an accompanying empirical version and implied data-augmentation paradigm. As a first sound justification of this proposal, we prove that this type of adaptive data-augmentation maintains consistency of a non-parametric method (namely 1-nearest neighbor classification under deterministic labels). We believe this to be a natural and useful take on resolving the discrepancies with accuracy that have been reported in the context of adversarial robustness (both in theoretical and practical studies). Further, we believe that our notion of a data-informed, adaptive robustness radius might be useful for other methods that employ data augmentation.

References

1 Naveed Akhtar and Ajmal Mian. Threat of adversarial attacks on deep learning in computer vision: A survey. *IEEE Access*, 6:14410–14430, 2018.

2 Hassan Ashtiani, Vinayak Pathak, and Ruth Urner. Black-box certification and learning under adversarial perturbations. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020*, 2020.

3 Idan Attias, Aryeh Kontorovich, and Yishay Mansour. Improved generalization bounds for robust learning. In *Algorithmic Learning Theory, ALT*, pages 162–183, 2019.

4 Pranjal Awasthi, Abhratanu Dutta, and Aravindan Vijayaraghavan. On robustness to adversarial examples and polynomial optimization. In *Advances in Neural Information Processing Systems, NeurIPS*, pages 13760–13770, 2019.

5 Yogesh Balaji, Tom Goldstein, and Judy Hoffman. Instance adaptive adversarial training: Improved accuracy tradeoffs in neural nets. *CoRR*, abs/1910.08051, 2019. `arXiv:1910.08051`.

6 Robi Bhattacharjee and Kamalika Chaudhuri. When are non-parametric methods robust? In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020*, 2020.

7 Robi Bhattacharjee and Kamalika Chaudhuri. Consistent non-parametric methods for adaptive robustness. *CoRR*, abs/2102.09086, 2021. `arXiv:2102.09086`.

8 Sébastien Bubeck, Yin Tat Lee, Eric Price, and Ilya P. Razenshteyn. Adversarial examples from computational constraints. In *Proceedings of the 36th International Conference on Machine Learning, ICML*, pages 831–840, 2019.

9 Nicholas Carlini, Anish Athalye, Nicolas Papernot, Wieland Brendel, Jonas Rauber, Dimitris Tsipras, Ian J. Goodfellow, Aleksander Madry, and Alexey Kurakin. On evaluating adversarial robustness. *CoRR*, abs/1902.06705, 2019. `arXiv:1902.06705`.

10 Anirban Chakraborty, Manaar Alam, Vishal Dey, Anupam Chattopadhyay, and Debdeep Mukhopadhyay. Adversarial attacks and defences: A survey. *CoRR*, abs/1810.00069, 2018. `arXiv:1810.00069`.

11 Kamalika Chaudhuri and Sanjoy Dasgupta. Rates of convergence for nearest neighbor classification. In *Advances in Neural Information Processing Systems, NIPS*, pages 3437–3445, 2014.

12 Jeremy M. Cohen, Elan Rosenfeld, and J. Zico Kolter. Certified adversarial robustness via randomized smoothing. In *Proceedings of the 36th International Conference on Machine Learning, ICML*, pages 1310–1320, 2019.

13 Daniel Cullina, Arjun Nitin Bhagoji, and Prateek Mittal. Pac-learning in the presence of adversaries. In *Advances in Neural Information Processing Systems, NeurIPS*, pages 230–241, 2018.

14 Gavin Weiguang Ding, Yash Sharma, Kry Yik Chau Lui, and Ruitong Huang. MMA training: Direct input space margin maximization through adversarial training. In *8th International Conference on Learning Representations, ICLR*, 2020.

15 Dimitrios Diochnos, Saeed Mahloujifar, and Mohammad Mahmoody. Adversarial risk and robustness: General definitions and implications for the uniform distribution. In *Advances in Neural Information Processing Systems 31, NeurIPS*, pages 10359–10368, 2018.

16 Uriel Feige, Yishay Mansour, and Robert Schapire. Learning and inference in the presence of corrupted inputs. In *Conference on Learning Theory, COLT*, pages 637–657, 2015.

17 Yarin Gal and Lewis Smith. Sufficient conditions for idealised models to have no adversarial examples: a theoretical and empirical study with bayesian neural networks, 2018. `arXiv:1806.00667`.

18 Ian J. Goodfellow, Patrick D. McDaniel, and Nicolas Papernot. Making machine learning robust against adversarial inputs. *Commun. ACM*, 61(7):56–66, 2018.

19 Pascale Gourdeau, Varun Kanade, Marta Kwiatkowska, and James Worrell. On the hardness of robust classification. In *Advances in Neural Information Processing Systems 32, NeurIPS*, pages 7444–7453, 2019.

20 Ruitong Huang, Bing Xu, Dale Schuurmans, and Csaba Szepesvári. Learning with a strong adversary. *CoRR*, abs/1511.03034, 2015. `arXiv:1511.03034`.

21 Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. In *Advances in Neural Information Processing Systems 32: NeurIPS*, pages 125–136, 2019.

22 Marc Khoury and Dylan Hadfield-Menell. Adversarial training with voronoi constraints. *CoRR*, abs/1905.01019, 2019. `arXiv:1905.01019`.

23 Samory Kpotufe. k-nn regression adapts to local intrinsic dimension. In *Advances in Neural Information Processing Systems, NIPS*, pages 729–737, 2011.

24 Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *6th International Conference on Learning Representations, ICLR*, 2018.

25 Omar Montasser, Surbhi Goel, Ilias Diakonikolas, and Nathan Srebro. Efficiently learning adversarially robust halfspaces with noise. *arXiv preprint*, 2020. `arXiv:2005.07652`.

26 Omar Montasser, Steve Hanneke, and Nathan Srebro. VC classes are adversarially robustly learnable, but only improperly. In *Conference on Learning Theory, COLT*, pages 2512–2530, 2019.

27 Chongli Qin, James Martens, Sven Gowal, Dilip Krishnan, Krishnamurthy Dvijotham, Alhussein Fawzi, Soham De, Robert Stanforth, and Pushmeet Kohli. Adversarial robustness through local linearization. In *Advances in Neural Information Processing Systems 32, NeurIPS*, pages 13824–13833, 2019.

28 Hadi Salman, Jerry Li, Ilya P. Razenshteyn, Pengchuan Zhang, Huan Zhang, Sébastien Bubeck, and Greg Yang. Provably robust deep learning via adversarially trained smoothed classifiers. In *Advances in Neural Information Processing Systems 32, NeurIPS*, pages 11289–11300, 2019.

29 Ludwig Schmidt, Shibani Santurkar, Dimitris Tsipras, Kunal Talwar, and Aleksander Madry. Adversarially robust generalization requires more data. In *Advances in Neural Information Processing Systems, NeurIPS*, pages 5014–5026, 2018.

30 Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014.

31 Ingo Steinwart and Clint Scovel. Fast rates for support vector machines using gaussian kernels. *The Annals of Statistics*, 35(2):575–607, 2007.

32 Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *2nd International Conference on Learning Representations, ICLR*, 2014.

33 Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. In *7th International Conference on Learning Representations, ICLR*, 2019.

34 Ruth Urner, Sharon Wulff, and Shai Ben-David. PLAL: cluster-based active learning. In *COLT 2013 - The 26th Annual Conference on Learning Theory*, pages 376–397, 2013.

35 Yizhen Wang, Somesh Jha, and Kamalika Chaudhuri. Analyzing the robustness of nearest neighbors to adversarial examples. In *Proceedings of the 35th International Conference on Machine Learning, ICML*, pages 5120–5129, 2018.

36 Huanrui Yang, Jingchi Zhang, Hsin-Pai Cheng, Wenhan Wang, Yiran Chen, and Hai Li. Bamboo: Ball-shape data augmentation against adversarial attacks from all directions. In *Workshop on Artificial Intelligence Safety 2019 co-located with the Thirty-Third AAAI Conference on Artificial Intelligence*, 2019.

37 Yao-Yuan Yang, Cyrus Rashtchian, Yizhen Wang, and Kamalika Chaudhuri. Adversarial examples for non-parametric methods: Attacks, defenses and large sample limits. *CoRR*, abs/1906.03310, 2019. `arXiv:1906.03310`.

38 Yao-Yuan Yang, Cyrus Rashtchian, Yizhen Wang, and Kamalika Chaudhuri. Robustness for non-parametric classification: A generic attack and defense. In *The 23rd International Conference on Artificial Intelligence and Statistics, AISTATS*, pages 941–951, 2020.

39 Yao-Yuan Yang, Cyrus Rashtchian, Hongyang Zhang, Russ R. Salakhutdinov, and Kamalika Chaudhuri. A closer look at accuracy vs. robustness. In *Advances in Neural Information Processing Systems 33 NeurIPS*, 2020.

40 Dong Yin, Kannan Ramchandran, and Peter L. Bartlett. Rademacher complexity for adversarially robust generalization. In *Proceedings of the 36th International Conference on Machine Learning,ICML*, pages 7085–7094, 2019.

41 Hang Yu, Aishan Liu, Xianglong Liu, Gengchao Li, Ping Luo, Ran Cheng, Jichen Yang, and Chongzhi Zhang. Pda: Progressive data augmentation for general robustness of deep neural networks, 2020. `arXiv:1909.04839`.

42 Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P. Xing, Laurent El Ghaoui, and Michael I. Jordan. Theoretically principled trade-off between robustness and accuracy. In *Proceedings of the 36th International Conference on Machine Learning, ICML*, pages 7472–7482, 2019.

A Visualizations

To further validate our proposed adaptive robust data augmentation method, we present a set of illustrative experiments on various synthetic datasets. To allow for visualizations, we generate data from a "lower-dimensional manifold" in two dimensions. It has been conjectured that the data being supported on a lower-dimensional manifold is a source of the phenomenon of vulnerability to small perturbations [22]. Our visualizations in in Figure 3 illustrate this phenomenon.

The original support (the data-manifold) of the data generating distributions is one-dimensional hehre and can be seen as the green and blue lines in the first column of images in Figure 3. Blue and green points represent points from the two classes. We term our synthetic shapes in Figure 3 **Sines, S-figure, NNN, circles, boxes**. We train a ReLU Neural Network with 2-hidden layers (of 10 neurons each) data points drawn from these shapes. The labeling behavior of the trained network is visualized over the ambient space in red and purple. The first image in each row depicts the original, labeled data together with the network trained on the original data.

We see in those left-most illustrations that without any augmentation, the network's decision boundary is often located close to the data-manifold. Since the data is supported only on the lower-dimensional manifold, there is no incentive for the decision boundary to keep a distance from the data-manifold. While the network labels areas on the manifold itself correctly, this behavior leads to the existence of points that are vulnerable to adversarial perturbations: a small deviation away from the data-manifold can lead to a different labeling by the network.

We then augment the training datasets with both fixed and adaptive expansion parameter and train ReLU Neural Networks of the same size on the augmented datasets. The remaining images in each row again illustrate the augmented datasets (green and blue) together with the labeling behaviors of the resulting networks (red and purple). The last image in each row corresponds to the adaptive augmented data, while the intermediate images correspond to augmentations with increasing, but fixed expansion parameters.

For non-adaptive expansion parameter, we iteratively increase the parameter in a fixed sequence, $(0.1, 0.5, 1, 2, ..., 16)$. These expansion parameters were chosen based on the range of the attribute values in the datasets. For each sample in a d-dimensional dataset, a d-dimensional sphere is generated where the radius is the fixed-parameter and the current sample is the center of the sphere. Four new points are then generated in this sphere for each sample point. Hence, the training dataset is expanded to five times its original size after fixed-parameter expansion.

Analogously we augment the data with an adaptive expansion parameter. The key difference is in the calculation of the radius of the sphere. A fraction of the distance between the current sample and a nearest neighbor of a different class is used as the radius for the sphere generation. Each of the middle columns in Figure 3 corresponds to augmentation with a fixed expansion parameter, while the last column shows the 2/3-adaptive robust augmentation of the training data. The original training dataset contains 1000 training points and the augmented datasets 5000 data points each.

For the various networks we evaluate binary loss and the adaptive robust loss. To estimate the adaptive robust loss at a point x, we determine its distance ρ to a point in the dataset with a different label and then generate 10 test points uniformly at random from a ball

of radius $\rho/2$. If one of these gets a different label than x by the network (or if the point is mislabeled itself) it suffers adaptive robust loss 1. Table 1 summarizes the binary and adaptive robust losses of the various networks. We see that the adaptive augmentation leads consistently to the lowest binary (always rank 1) and low adaptive robust loss (rank 1 and once rank 2). This shows that the adaptive augmentation not only is not in conflict with accuracy, but empirically improves accuracy of a trained network.

Finally, we also trained ReLU neural networks on several real-world data sets from the UCI repository. For each dataset, we normalized the features to take values in [0,1]. As in the experiments on the synthetic data, we trained the networks on the original data, as well as various augmented datasets, including using the 2/3-adaptive augmentation. The datasets were split into training and test data with a ratio of $80 - 20$ respectively. In Tables 1 and 2, we report the binary and robust losses of these networks. We observe, again, that the robust

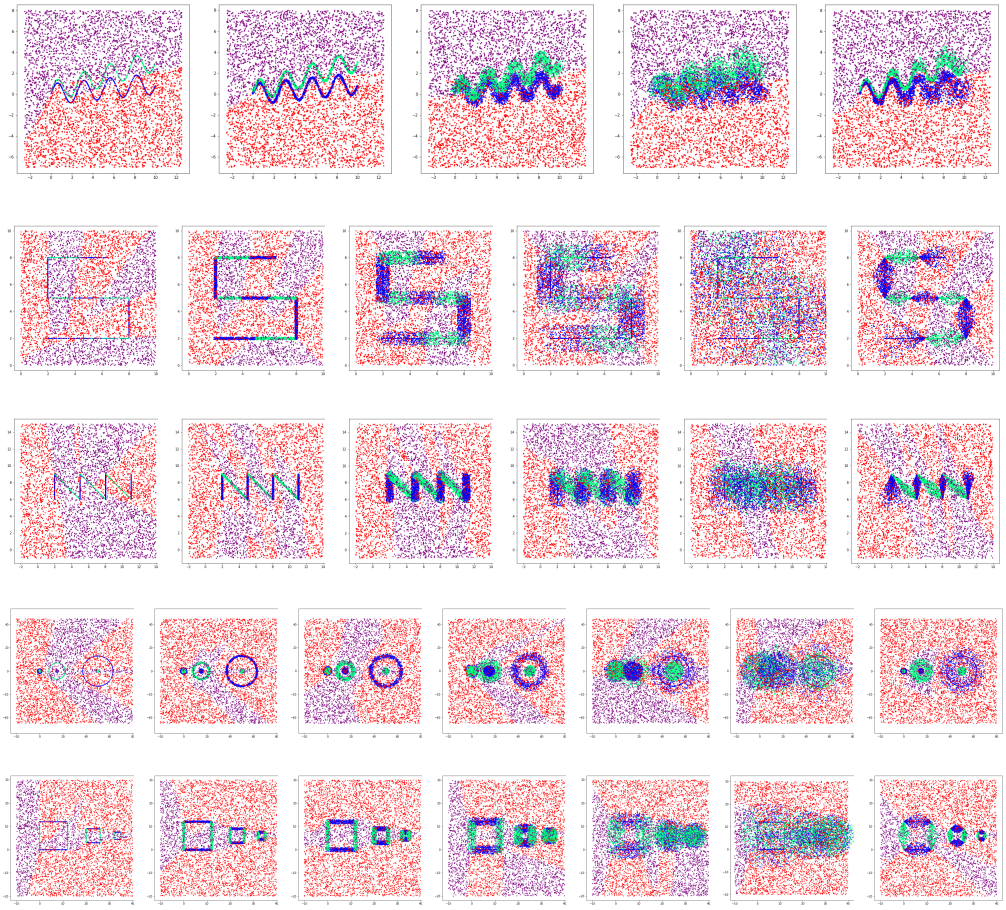

Figure 3 ReLU networks trained on data from a one-dimensional manifold in two-dimensional space, labeled using two classes (blue and green here). The various shapes by row: **Sines, S-figure, NNN, circles, boxes**. Left-most: original training data; various middle images: training data augmented using increasing expansion parameters; right-most: training data robust-adaptive expanded. We use data generated uniformly at random from the ambient space to illustrate the network's labeling (red and purple). Using just original training data, or only slightly augmented data, we observe that the network's decision boundary is often close to the manifold.

augmentation promotes the best performance in terms of 0/1 accuracy. Additionally, the adaptive robust loss is close to the best adaptive robust loss achieved with a fixed expansion parameter on each dataset. Using the adaptive augmentation can thus serve to save needing to search for an optimal expansion parameter on different tasks.

In summary, our initial experimental explorations here showed that the adaptive augmentation consistently yielded a robust predictor with best 0/1-loss. This confirms the intended design of an adaptive robustness and data augmentation paradigm that avoids the undesirable tradeoffs between robustness and accuracy.

Table 1 Overview on the binary and adaptive robust losses of the networks trained on trained on the various synthetic datasets with various augmentations.

Dataset	Expansion Radius for Training	Adaptive Robust Loss	Binary Loss
Sines	Original	0.2882	0.104
	0.1	0.1693	0.071
	0.5	0.2443	0.147
	1	0.3116	0.177
	2	0.3521	0.208
	Adaptive	**0.1403**	**0.038**
S-figure	Original	0.3516	0.044
	0.1	0.1514	0.016
	0.5	0.0429	0.027
	1	0.0844	0.05
	2	0.2373	0.21
	Adaptive	**0.0393**	**0.017**
NNN	Original	0.3841	0.2124
	0.1	0.2609	0.1086
	0.5	0.2008	0.1048
	1	0.1969	0.0952
	2	0.386	0.3714
	Adaptive	**0.08972**	**0.04**
circles	Original	0.4483	0.0133
	0.5	0.2629	**0**
	1	0.3472	0.0108
	2	**0.1778**	0.0242
	4	0.3076	0.0783
	8	0.3557	0.1733
	16	0.3054	0.1633
	Adaptive	0.254	**0**
boxes	Original	0.3427	0.08
	0.5	0.2623	0.0775
	1	0.2229	0.0775
	2	0.2252	0.1667
	4	0.2839	0.2283
	8	0.4274	0.3458
	Adaptive	**0.2077**	**0.075**

Table 2 Overview on the binary and adaptive robust losses of the networks trained on trained on the various UCI datasets with various augmentations.

Dataset	Expansion Radius for Training	Adaptive Robust Loss	Binary Loss
Iris	Original	0.0957	0.0435
	0.1	**0.0783**	**0**
	0.5	0.1304	**0**
	1	0.3478	0.087
	2	0.391	0.3478
	Adaptive	0.087	**0**
Breast Cancer	Original	0.1351	0.0263
	0.1	0.0956	**0.0175**
	0.5	0.0842	0.0351
	1	0.0833	0.0439
	2	**0.0693**	**0.0175**
	Adaptive	0.0719	**0.0175**
Bank Note Authentication	Original	0.0804	**0**
	0.1	0.0479	**0**
	0.5	0.1593	0.0909
	1	0.1153	0.0036
	2	0.1058	0.0036
	Adaptive	**0.0167**	**0**
Heart Disease	Original	**0.3465**	0.1628
	0.1	0.3791	0.2093
	0.5	0.386	0.2093
	1	0.4489	0.2791
	2	0.507	0.3488
	Adaptive	0.3604	**0.1395**
Immunotherapy	Original	0.263	0.1852
	0.1	0.2926	0.1111
	0.5	0.3482	0.1852
	1	0.2333	0.1852
	2	0.437	0.2593
	Adaptive	**0.174**	**0.0741**
Parkinsons	Original	**0.1423**	0.078
	0.1	0.1678	0.0847
	0.5	0.1542	0.0678
	1	0.2322	0.1017
	2	0.2322	0.1186
	Adaptive	0.1627	**0.0508**