UNIVERSITY OF CALIFORNIA SAN DIEGO

The Primacy of Applied Privacy

A dissertation submitted in partial satisfaction of the
requirements for the degree Doctor of Philosophy

in

Computer Science

by

Casey Meehan

Committee in charge:

Professor Kamalika Chaudhuri, Chair
Professor Taylor Berg-Kirkpatrick
Professor Sanjoy Dasgupta
Professor Alon Orlitsky

2023

The Dissertation of Casey Meehan is approved, and it is acceptable in quality and form for publication on microfilm and electronically.

University of California San Diego

2023

DEDICATION

The fact that I have made something I can write a dedication for is owed all to my parents. I cannot imagine following my heart these past few years without their unrelenting support and encouragement.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

VITA

2015	Bachelor of Science, Brown University
2018	Master of Science, Harvard University
2023	Doctor of Philosophy, University of California, San Diego

ABSTRACT OF THE DISSERTATION

The Primacy of Applied Privacy

by

Casey Meehan

Doctor of Philosophy in Computer Science

University of California San Diego, 2023

Professor Kamalika Chaudhuri, Chair

As data collection for machine learning (ML) tasks has become more pervasive, it has also become more heterogeneous: we share our writing, images, voices, and location online every day. Naturally, the associated privacy risks are just as complex and variable. My research advances practical data privacy through two avenues: 1) drafting provable privacy definitions and mechanisms for safely sharing data in different ML domains, and 2) empirically quantifying how ML models memorize their sensitive training data and thereby risk disclosing it. This dissertation details the various data domains/tasks considered, and the corresponding privacy methods proposed.

Chapter 1

When are Non-Parametric Methods Robust?

1.1 Introduction

Recent work has shown that many classifiers tend to be highly non-robust and that small strategic modifications to regular test inputs can cause them to misclassify [1, 2, 3]. Motivated by the use of machine learning in safety-critical applications, this phenomenon has recently received considerable interest; however, what exactly causes this phenomenon – known in the literature as *adversarial examples* – still remains a mystery.

Prior work has looked at three plausible reasons why adversarial examples might exist. The first, of course, is the possibility that in real data distributions, different classes are very close together in space – which does not seem plausible in practice. Another possibility is that classification algorithms may require more data to be robust than to be merely accurate; some prior work [4, 5, 6] suggests that this might be true for certain classifiers or algorithms. Finally, others [7, 8, 5] have suggested that better training algorithms may give rise to more robust classifiers – and that in some cases, finding robust classifiers may even be computationally challenging.

In this work, we consider this problem in the context of general non-parametric classifiers. Contrary to parametrics, non-parametric methods are a form of local classifiers, and include a large number of pattern recognition methods such as nearest neighbors, decision trees, random

forests and kernel classifiers. There is a richly developed statistical theory of non-parametric methods [9], which focuses on accuracy, and provides very general conditions under which these methods converge to the Bayes optimal with growing number of samples. We, in contrast, analyze robustness properties of these methods, and ask instead when they converge to the classifier with the highest astuteness at a desired radius r. Recall that the astuteness of a classifier at radius r is the fraction of points from the distribution on which it is accurate and has the same prediction up to a distance r [5, 4].

We begin by looking at the very simple case when data from different classes is well-separated – by at least a distance $2r$. Although achieving astuteness in this case may appear trivial, we show that even in this highly favorable case, not all non-parametric methods provide robust classifiers – and this even holds for methods that converge to the Bayes optimal in the large sample limit.

This raises the natural question – when do non-parametric methods produce astute classifiers? We next provide conditions under which a non-parametric method converges to the most astute classifier in the large sample limit under well-separated data. Our conditions are analogous to the classical conditions for convergence to the Bayes optimal [9, 10], but a little stronger. We show that nearest neighbors and kernel classifiers whose kernel functions decay fast enough, satisfy these conditions, and hence converge to astute classifiers in the large sample limit. In constrast, histogram classifiers, which do converge to the Bayes optimal in the large sample limit, may not converge to the most astute classifier. This indicates that there may be some non-parametric methods, such as nearest neighbors and kernel classifiers, that are more naturally robust when trained on well-separated data, and some that are not.

What happens when different classes in the data are not as well-separated? For this case, [11] proposes a method called Adversarial Pruning that preprocesses the training data by retaining the maximal set of points such that different classes are distance $\geq 2r$ apart, and then trains a non-parametric method on the pruned data. We next prove that if a non-parametric method has certain properties, then the classifier produced by Adversarial Pruning followed by

the method does converges to the most astute classifier in the large sample limit. We show that again nearest neighbors and kernel classifiers whose kernel functions decay faster than inverse polynomials satisfy these properties. Our results thus complement and build upon the empirical results of [11] by providing a performance guarantee.

What can we conclude about the cause for adversarial examples? Our results seem to indicate that at least for non-parametrics, it is mostly the training algorithms that are responsible. With a few exceptions, decades of prior work in machine learning and pattern recognition has largely focussed on designing training methods that provide increasingly accurate classifiers – perhaps to the detriment of other aspects such as robustness. In this context, our results serve to (a) provide a set of guidelines that can be used for designing non-parametric methods that are robust and accurate on well-separated data and (b) demonstrate that when data is not well-separated, preprocessing through adversarial pruning [11] may be used to ensure convergence to optimally astute solutions in the large sample limit.

1.1.1 Related Work

There is a large body of work on adversarial attacks [12, 13, 14, 15, 1] and defenses [16, 17, 18, 19, 20, 21] in the parametric setting, specifically focusing on neural networks. On the other hand, adversarial examples for nonparametric classifiers have mostly been studied in a much more ad-hoc manner, and to our knowledge, there has been no theoretical investigation into general properties of algorithms that promote robustness in non-parametric classifiers.

For nearest neighbors, there has been some prior work on adversarial attacks [22, 23, 5, 11] as well as defenses. Wang et. al. [5] proposes a defense for 1-NN by pruning the input sample. However, their defense learns a classifier whose robustness regions converge towards those of the Bayes optimal classifier, which itself may potentially have poor robustness properties. Yang et. al. [11] accounts for this problem by proposing the notion of the r-optimal classifier, and propose an algorithm called Adversarial Pruning which can be interpreted as a finite sample approximation to the r-optimal. However, they do not provide formal performance guarantees

for Adversarial Pruning, which we do.

For Kernel methods, Hein and Andriushchenko [16] study lower bounds on the norm of the adversarial manipulation that is required for changing a classifiers output. They specifically study bounds for Kernel Classifiers, and propose an empirically based regularization idea that improves robustness. In this work, we improve the robustness properties of kernel classification through adversarial pruning, and show formal guarantees regarding convergence towards the r-optimal classifier.

For decision trees and random forests, attacks and defenses have been provided by [24, 25, 26]. Again, most of the work here is empirical in nature, and convergence guarantees are not provided.

Pruning has a long history of being applied for improving nearest neighbors [27, 28, 29, 30, 31, 32], but this has been entirely done in the context of generalization, without accounting for robustness. In their work, Yang et. al. empirically show that adversarial pruning can improve robustness for nearest neighbor classifiers. However, they do not provide any formal guarantees for their algorithms. In this work, we prove formal guarantees for *adversarial pruning* in the large sample limit, both for nearest neighbors as well as for more general *weight functions.*

There is a long history of literature for understanding the consistency of Kernel classifiers [33, 10], but this has only been done for accuracy and generalization. In this work, we find different conditions are needed to ensure that a Kernel classifier converges in robustness in addition to accuracy.

1.2 Preliminaries

1.2.1 Setting

We consider binary classification where instances are drawn from a totally bounded metric space $\mathscr{X}$ that is equipped with distance metric denoted by d, and the label space is $\{\pm 1\} = \{-1, +1\}$. The classical goal of classification is to build a highly *accurate* classifier,

which we define as follows.

Definition 1. *(Accuracy) Let $\mathscr{D}$ be a distribution over $\mathscr{X} \times \{\pm 1\}$, and let $f \in \{\pm 1\}^{\mathscr{X}}$ be a classifier. Then the* ***accuracy*** *of f over $\mathscr{D}$, denoted $A(f, \mathscr{D})$, is the fraction of examples $(x, y) \sim \mathscr{D}$ for which $f(x) = y$. Thus*

$$A(f, \mathscr{D}) = P_{(x,y)\sim\mathscr{D}}[f(x) = y].$$

In this work, we consider *robustness* in addition to accuracy. Let $B(x, r)$ denoted the closed ball of radius r centered at x.

Definition 2. *(Robustness) A classifier $f \in \{\pm 1\}^{\mathscr{X}}$ is said to be* ***robust*** *at x with radius r if $f(x) = f(x')$ for all $x' \in B(x, r)$.*

Our goal is to find non-parametric algorithms that output classifiers that are robust, in addition to being accurate. To account for both criteria, we combine them into a notion of *astuteness* [5, 4].

Definition 3. *(Astuteness) A classifier $f \in \{\pm 1\}^{\mathscr{X}}$ is said to be* ***astute*** *at (x, y) with radius r if f is robust at x with radius r and $f(x) = y$. The* ***astuteness*** *of f over $\mathscr{D}$, denoted $A_r(f, \mathscr{D})$, is the fraction of examples $(x, y) \sim \mathscr{D}$ for which f is astute at (x, y) with radius r. Thus*

$$A_r(f, \mathscr{D}) = P_{(x,y)\sim\mathscr{D}}[f(x') = y, \forall x' \in B(x, r)].$$

It is worth noting that $A_0(f, \mathscr{D}) = A(f, \mathscr{D})$, since astuteness with radius 0 is simply the accuracy. For this reason, we will use $A_0(f, \mathscr{D})$ to denote accuracy from this point forwards.

1.2.2 Notions of Consistency

Traditionally, a classification algorithm is said to be consistent if as the sample size grows to infinity, the accuracy of the classifier it learns converges towards the best possible accuracy on the underlying data distribution. We next introduce and formalize an alternative form of consistency, called r-consistency, that applies to robust classifiers.

We begin with a formal definition of the Bayes Optimal Classifier – the most accurate classifier on a distribution – and consistency.

Definition 4. *(Bayes Optimal Classifier) The* ***Bayes Optimal Classifier*** *on a distribution $\mathscr{D}$, denoted by g^*, is defined as follows. Let $\eta(x) = p_{\mathscr{D}}(+1|x)$. Then*

$$g^*(x) = \begin{cases} +1 & \eta(x) \geq 0.5 \\ -1 & \eta(x) < 0.5 \end{cases}$$

It can be shown that g^* achieves the highest accuracy over $\mathscr{D}$ over all classifiers.

Definition 5. *(Consistency) Let M be a classification algorithm over $\mathscr{X} \times \{\pm 1\}$. M is said to be* ***consistent*** *if for any $\mathscr{D}$ over $\mathscr{X} \times \{\pm 1\}$, and any ε, δ over $(0,1)$, there exists N such that for $n \geq N$, with probability $1-\delta$ over $S \sim \mathscr{D}^n$, we have:*

$$A(M(S), \mathscr{D}) \geq A(g^*, \mathscr{D}) - \varepsilon,$$

where g^* is the Bayes optimal classifier for $\mathscr{D}$.

How can we incorporate robustness in addition to accuracy in this notion? A plausible way, as used in [5], is that the classifier should converge towards being astute where the Bayes Optimal classifier is astute. However, the Bayes Optimal classifier is not necessarily the most astute classifier and may even have poor astuteness. To see this, consider the following example.

Example 1

Consider $\mathscr{D}$ over $\mathscr{X} = [0,1]$ such that $\mathscr{D}_{\mathscr{X}}$ is the uniform distribution and

$$p(y=1|x) = \frac{1}{2} + \sin\frac{4\pi x}{r}.$$

For any point x, there exists $x_1, x_2 \in ([x-r, x+r] \cap [0,1])$ such that $p(y=1|x_1) > \frac{1}{2}$ and $p(y=1|x_2) < \frac{1}{2}$. $A_r(g^*, r) = 0$. However, the classifier that always predicts $f(x) = +1$ does

better. It is robust everywhere, and since $P_{(x,y)\sim\mathscr{D}}[y=+1]=\frac{1}{2}$, it follows that $A_r(f,\mathscr{D})=\frac{1}{2}$.

This motivates the notion of the r-optimal classifier, introduced by [11], which is the classifier with maximum astuteness.

Definition 6. *(r-optimal classifier) The* ***r-optimal classifier*** *of a distribution G denoted by g_r^* is the classifier with maximum astuteness. Thus*

$$g_r^* = \underset{f\in\{\pm1\}^{\mathscr{X}}}{\arg\max}\, A_r(f,\mathscr{D}).$$

We let $A_r^*(\mathscr{D})$ denote $A_r(g_r^*,\mathscr{D})$.

Observe that g_r^* is not necessarily unique. To account for this, we use $A_r^*(\mathscr{D})$ in our definition for r-consistency.

Definition 7. *(r-consistent) Let M be a classification algorithm over $\mathscr{X}\times\{\pm1\}$. M is said to be* ***r-consistent*** *if for any $\mathscr{D}$, any $\varepsilon,\delta\in(0,1)$, and $0<\gamma<r$, there exists N such that for $n\geq N$, with probability $1-\delta$ over $S\sim\mathscr{D}^n$,*

$$A_{r-\gamma}(M(S),\mathscr{D})\geq A_r^*(\mathscr{D})-\varepsilon.$$

if the above conditions hold for a specific distribution $\mathscr{D}$, we say that M is r-consistent with respect to $\mathscr{D}$.

Observe that in addition to the usual ε and δ, there is an extra parameter γ which measures the gap in the robustness radius. We may need this parameter as when classes are exactly $2r$ apart, we may not be able to find the exact robust boundary with only finite samples.

Our analysis will be centered around understanding what kinds of algorithms M provide highly astute classifiers for a given radius r. We begin by first considering the special case of

r-separated distributions.

Definition 8. *(r-separated distributions) A distribution $\mathscr{D}$ is said to be* ***r-separated*** *if there exist subsets $T^+, T^- \subset \mathscr{X}$ such that*

1. $\mathbb{P}_{(x,y)\sim\mathscr{D}}[x \in T^y] = 1$.
2. $\forall x_1 \in T^+, \forall x_2 \in T^-$, $d(x_1, x_2) > 2r$.

Observe that if $\mathscr{D}$ is r-separated, $A_r(g_r^*, \mathscr{D}) = 1$.

1.2.3 Non-parametric Classifiers

Many non-parametric algorithms classify points by averaging labels over a local neighborhood from their training data. A very general form of this idea is encapsulated in *weight functions* – which is the general form we will use.

Definition 9. *[9] A* ***weight function*** *W is a non-parametric classifier with the following properties.*

1. *Given input $S = \{(x_1, y_1), (x_2, y_2,), \ldots, (x_n, y_n)\} \sim \mathscr{D}^n$, W constructs functions $w_1^S, \ldots, w_n^S : \mathscr{X} \to [0,1]$ such that for all $x \in \mathscr{X}$, $\sum_1^n w_i^S(x) = 1$. The functions w_i^S are allowed to depend on $x_1, x_2, \ldots x_n$ but must be independent of $y_1, y_2, \ldots, y_n$.*
2. *W has output W_S defined as*

$$W_S(x) = \begin{cases} +1 & \sum_1^n w_i^S(x) y_i > 0 \\ -1 & \sum_1^n w_i^S(x) y_i \leq 0 \end{cases}$$

As a result, $w_i^S(x)$ can be thought of as the weight that (x_i, y_i) has in classifying x.

Weight functions encompass a fairly extensive set of common non-parametric classifiers, which is the motivation for considering them. We now define several common non-parametric algorithms that can be construed as weight functions.

Definition 10. *A **histogram classifier**, H, is a non-parametric classification algorithm over $\mathbb{R}^d \times \{\pm 1\}$ that works as follows. For a distribution $\mathscr{D}$ over $\mathbb{R} \times \{\pm 1\}$, H takes $S = \{(x_i, y_i) : 1 \leq i \leq n\} \sim \mathscr{D}^n$ as input. Let k_i be a sequence with $\lim_{i\to\infty} k_i = \infty$ and $\lim_{i\to\infty} \frac{k_i}{i} = 0$. H constructs a set of hypercubes $C = \{c_1, c_2, \ldots, c_m\}$ as follows:*

1. *Initially $C = \{c\}$, where $S \subset c$.*
2. *For $c \in C$, if c contains more than k_n points of S, then partition c into 2^d equally sized hypercubes, and insert them into C.*
3. *Repeat step 2 until all cubes in C have at most k_n points.*

For $x \in \mathbb{R}$ let $c(x)$ denote the unique cell in C containing x. If $c(x)$ doesn't exist, then $H_S(x) = -1$ by default. Otherwise,

$$H_S(x) = \begin{cases} +1 & \sum_{x_i \in c(x)} y_i > 0 \\ -1 & \sum_{x_i \in c(x)} y_i \leq 0 \end{cases}.$$

Histogram classifiers are weight functions in which all x_i contained within the same cell as x are given the same weight $w_i^S(x)$ in predicting x, while all other x_i are given weight 0.

Definition 11. *A **kernel classifier** is a weight function W over $\mathscr{X} \times \{\pm 1\}$ constructed from function $K : \mathbb{R}^+ \cup \{0\} \to \mathbb{R}^+$ and some sequence $\{h_n\} \subset \mathbb{R}^+$ in the following manner. Given $S = \{(x_i, y_i)\} \sim \mathscr{D}^n$, we have*

$$w_i^S(x) = \frac{K(\frac{d(x,x_i)}{h_n})}{\sum_{j=1}^n K(\frac{d(x,x_j)}{h_n})}.$$

Then, as above, W has output

$$W_S(x) = \begin{cases} +1 & \sum_1^n w_i^S(x) y_i > 0 \\ -1 & \sum_1^n w_i^S(x) y_i \leq 0 \end{cases}$$

Finally, we note that k_n-nearest neighbors is also a weight function; $w_i^S(x) = \frac{1}{k_n}$ if x_i is one of the k_n closest neighbors of x and 0 otherwise.

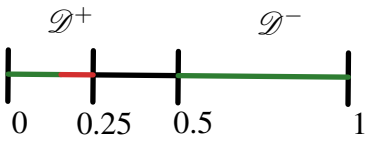


Figure 1.1. H_S is astute in the green region, but not robust in the red region.

1.3 Warm Up: r-separated distributions

We begin by considering the case when the data distribution is r-separated; the more general case is considered in Section 1.4. While classifying r-separated distributions robustly may appear almost trivial, learning an arbitrary classifier does not necessarily produce an astute result. To see this, consider the following example of a histogram classifier – which is known to be consistent.

We let H denote the histogram classifier over $\mathbb{R}$.

Example 2

Consider the data distribution $\mathscr{D} = \mathscr{D}^+ \cup \mathscr{D}^-$ where D^+ is the uniform distribution over $[0, \frac{1}{4})$ and D^- is the uniform distribution over $(\frac{1}{2}, 1]$, $p(+1|x) = 1$ for $x \in \mathscr{D}^+$, and $p(-1|x) = 1$ for $x \in \mathscr{D}^-$.

We make the following observations (refer to Figure 1.1).

1. $\mathscr{D}$ is 0.1-separated, since the supports of $\mathscr{D}^+$ and $\mathscr{D}^-$ have distance $0.25 > 0.2$.

2. If n is sufficiently large, H will construct the cell $[0.25, 0.5)$, which will not be split because it will never contain any points.

3. $H_S(x) = -1$ for $x \in [0.25, 0.5)$.

4. H_S is not astute at $(x, 1)$ for $x \in (0.15, 0.25)$. Thus $A_{0.1}(H_S, \mathscr{D}) = 0.8$.

Example 2 shows that histogram classifiers do not always learn astute classifiers even

when run on r-separated distributions. This motivates the question: which non-parametric classifiers do?

We answer this question in the following theorem, which gives sufficient conditions for a weight function (definition 9) to be r-consistent over an r-separated distribution.

Theorem 12. *Let $\mathscr{D}$ be a distribution over $\mathscr{X} \times \{\pm 1\}$, and let W be a weight function. Let X be a random variable with distribution $\mathscr{D}_{\mathscr{X}}$, and $S = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\} \sim \mathscr{D}^n$. Suppose that for any $0 < a < b$,*

$$\lim_{n\to\infty} \mathbb{E}_{X,S}\big[\sup_{x' \in B(X,a)} \sum_{1}^{n} w_i^S(x') I_{||x_i - x'|| > b}\big] = 0.$$

Then if $\mathscr{D}$ is r-separated, W is r-consistent with respect to $\mathscr{D}$.

First, we compare Theorem 12 to Stone's theorem [10], which gives sufficient conditions for a weight function to be consistent (i.e. converge in accuracy towards the Bayes optimal). For convenience, we include a statement of Stone's theorem.

Theorem 13. *[10] Let W be weight function over $\mathscr{X} \times \{\pm 1\}$. Suppose the following conditions hold for any distribution $\mathscr{D}$ over $\mathscr{X} \times \{\pm 1\}$. Let X be a random variable with distribution $\mathscr{D}_{\mathscr{X}}$, and $S = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\} \sim \mathscr{D}^n$. All expectations are taken over X and S.*

1. *There is a constant c such that, for every nonnegative measurable function f satisfying $\mathbb{E}[f(X)] < \infty$,*

$$\mathbb{E}[\sum_{1}^{n} w_i^S(X) f(x_i)] \leq c\mathbb{E}[f(x)].$$

2. *For all $a > 0$,*

$$\lim_{n\to\infty} \mathbb{E}[\sum_{1}^{n} w_i^S(x) I_{||x_i - X|| > a}] = 0,$$

   where $I_{||x_i - X|| > a}$ is an indicator variable.

3.

$$\lim_{n\to\infty} \mathbb{E}[\max_{1 \leq i \leq n} w_i^S(X)] = 0.$$

Then W is consistent.

There are two main differences between Theorem 12 and Stone's theorem.

1. Conditions 1. and 3. of Stone's theorem are no longer necessary. This is because r-separated distributions are well-separated and thus have simpler conditions for consistency. In fact, a slight modification of the arguments of [10] shows that for r-separated distributions, condition 2. alone is sufficient for consistency.

2. Condition 2. is strengthened. Instead of requiring the weight of x_i's outside of a given radius to go to 0 for $X \sim \mathscr{D}$, we require the same to *uniformly* hold over a ball centered at X.

Theorem 12 provides a general condition that allows us to verify the r-consistency of non-parametric methods. We now show below that two common non-parametric algorithms – k_n-nearest neighbors and kernel classifiers with rapidly decaying kernel functions – satisfy the conditions of Theorem 12.

Corollary 14. *Let $\mathscr{D}$ be any r-separated distribution. Let k_n be any sequence such that $\lim_{n\to\infty} \frac{k_n}{n} = 0$, and let M be the k_n-nearest neighbors classifier on a sample $S \sim \mathscr{D}^n$. Then M is r-consistent with respect to $\mathscr{D}$.*

Remarks:

1. Because the data distribution is r-separated, $k_n = 1$ will be r-consistent. Also observe that for r-separated distributions, $k_n = 1$ will converge towards the Bayes Optimal classifier.

2. In general, M converges towards the Bayes Optimal classifier provided that $k_n \to \infty$ in addition to $k_n/n \to 0$. This condition is not necessary for r-consistency– because the distribution is r-separated.

We next show that kernel classifiers are also r-consistent on r-separated data distributions, provided the kernel function decreases rapidly enough.

Corollary 15. *Let W be a kernel classifier over $\mathscr{X} \times \{\pm 1\}$ constructed from K and h_n. Suppose the following properties hold for K and h_n.*

1. *For any $c > 1$, $\lim_{x\to\infty} \frac{K(cx)}{K(x)} = 0$.*
2. *$\lim_{n\to\infty} h_n = 0$.*

If $\mathscr{D}$ is an r-separated distribution over $\mathscr{X} \times \{\pm 1\}$, then W is r-consistent with respect to $\mathscr{D}$.

Observe that Condition 1. is satisfied for any $K(x)$ that decreases more rapidly than an inverse polynomial – and is hence satisfied by most popular kernels like the Gaussian kernel. Is the condition on K in Corollary 15 necessary? The following example illustrates that a kernel classifier with any arbitrary K is not necessarily r-consistent. This indicates that some sort of condition needs to be imposed on K to ensure r-consistency; finding a tight necessary condition however is left for future work.

Example 3

Let $\mathscr{X} = [-1,1]$ and let $\mathscr{D}$ be a distribution with $p_{\mathscr{D}}(-1,-1) = 0.1$ and $p_{\mathscr{D}}(1,1) = 0.9$. Clearly, $\mathscr{D}$ is 0.3-separated. Let $K(x) = e^{-\min(|x|,0.2)^2}$. Let h_n be any sequence with $\lim_{n\to\infty} h_n = 0$ and $\lim_{n\to\infty} nh_n = \infty$. Let W be the weight classifier with input $S = \{(x_1,y_1),(x_2,y_2),\ldots,(x_n,y_n)\}$ such that

$$w_i^S(x) = \frac{K(\frac{|x-x_i|}{h_n})}{\sum_{j=1}^n K(\frac{|x-x_j|}{h_n})}.$$

W can be shown to satisfy all the conditions of Theorem 13 (the proof is analogous to the case for a Gaussian Classifier), and is therefore consistent. However, W does not learn a robust classifier on $\mathscr{D}$ for $r = 0.3$.

Consider $x = -0.7$. For any $\{(x_1,y_1),(x_2,y_2),\ldots,(x_n,y_n)\} \sim \mathscr{D}^n$, all x_i will either be -1 or 1. Therefore, since $K(|x-(-1)|) = K(|x-1|)$, it follows that $w_i^S(x) = \frac{1}{n}$ for all $1 \le i \le n$. Since $x_i = 1$ with probability 0.9, it follows that with high probability x will be classified as 1 which means that f, the output of W, is not robust at $x = -1$. Thus f has astuteness at most 0.9 which means that W is *not* r-consistent for $r = 0.3$.

1.4 General Distributions

We next consider more general data distributions, where data from different classes may be close together in space, and may even overlap. Observe that unlike the r-separated case, here there may be no classifier with astuteness one. Thus, a natural question is: what does the optimally astute classifier look like, and how can we build non-parametric classifiers to this limit?

1.4.1 The r-Optimal Classifier and Adversarial Pruning

[11] propose a large-sample limit – called the r-optimal – and show that it is analogous to the Bayes Optimal classifier for robustness. More specifically, given a data distribution D, to find the r-optimal classifier, we solve the following optimization problem.

$$\begin{aligned} \max_{S_{+1},S_{-1}} & \int_{x\in S_{+1}} p(y=+1|x)d\mu_{\mathscr{D}}(x)+ \\ & \int_{x\in S_{-1}} p(y=-1|x)d\mu_{\mathscr{D}}(x) \\ & \text{subject to } d(S_{+1},S_{-1})>2r \end{aligned} \tag{1.1}$$

Then, the r-optimal classifier is defined as follows.

Definition 16. *[11] Fix $r,\mathscr{D}$. Let S^*_{+1} and S^*_{-1} be any optimizers of (1.1). Then the r-optimal classifier, g^*_r is any classifier such that $g^*_r(x)=j$ whenever $d(S^*_j,x)\leq r$.*

[11] show that the r-optimal classifier achieves the optimal astuteness – out of all classifiers on the data distribution $\mathscr{D}$; hence, it is a robustness analogue to the Bayes Optimal Classifier. Therefore, for general distributions, the goal in robust classification is to find non-parametric algorithms that output classifiers that converge towards g^*_r.

To find robust classifiers, [11] propose Adversarial Pruning – a defense method that preprocesses the training data by making it better separated. More specifically, Adversarial

Pruning takes as input a training dataset S and a radius r, and finds the largest subset of the training set where differently labeled points are at least distance $2r$ apart.

Definition 17. *A set $S_r \subset \mathscr{X} \times \{\pm 1\}$ is said to be **r-separated** if for all $(x_1, y_1), (x_2, y_2) \in S_r$, if $y_1 \neq y_2$, then $d(x_1, x_2) > 2r$. To **adversarially prune** a set S is to return its largest r-separated subset. We let $AdvPrun(S, r)$ denote the result of adversarially pruning S.*

Once an r-separated subset S_r of the training set is found, a standard non-parametric method is trained on S_r. While [11] show good empirical performance of such algorithms, no formal guarantees are provided. We next formally characterize when adversarial pruning followed by a non-parametric method results in a classifier that is provably r-consistent.

Specifically, we consider analyzing the general algorithm provided in Algorithm 1.

Algorithm 1: RobustNonPar

Input: $S \sim \mathscr{D}^n$, weight function W, robustness radius r;

$S_r \leftarrow AdvPrun(S, r)$;

Output: W_{S_r};

1.4.2 Convergence Guarantees

We begin with some notation. For any weight function W and radius $r > 0$, we let $RobustNonPar(W, r)$ represent the weight function that outputs weights for $S \sim \mathscr{D}^n$ according to $RobustNonPar(S, W, r)$. In particular, this can be used to convert any weight function algorithm into a new weight function which takes robustness into account. A natural question is, for which weight functions W is $RobustNonPar(W, r)$ r-consistent? Our next theorem provides sufficient conditions for this.

Theorem 18. *Let W be a weight function over $\mathscr{X} \times \{\pm 1\}$, and let $\mathscr{D}$ be a distribution over $\mathscr{X} \times \{\pm 1\}$. Fix $r > 0$. Let $S_r = AdvPrun(S, r)$. For convenience, relabel x_i, y_i so that $S_r =$*

$\{(x_1,y_1),(x_2,y_2),\ldots,(x_m,y_m)\}$. *Suppose that for any* $0<a<b$,

$$\lim_{n\to\infty}\mathbb{E}_{S\sim\mathscr{D}^n}\Big[\frac{1}{m}\sum_{i=1}^{m}\sup_{x\in B(x_i,a)}\sum_{j=1}^{m}w_j^{S_r}(x)I_{||x_j-x||>b}\Big]=0.$$

Then RobustNonPar(W,r) *is r-consistent with respect to* $\mathscr{D}$.

Remark:

There are two important differences between the conditions in Theorem 18 and Theorem 12.

1. We replace S with S_r.

2. The expectation over $X\sim\mathscr{D}_{\mathscr{X}}$ is replaced with an average over $\{x_1,x_2,\ldots,x_m\}$. The intuition here is that we are replacing $\mathscr{D}$ with a uniform distribution over S_r. While $\mathscr{D}$ may not be r-separated, the uniform distribution over S_r is, and represents the region of points where our classifier is astute.

A natural question is what satisfies the conditions in Theorem 18. We next show that k_n-nearest neighbors and kernel classifiers with rapidly decaying kernel functions continue to satisfy the conditions in Theorem 18; this means that these classifiers, when combined with Adversarial Pruning, will converge to r-optimal classifiers in the large sample limit.

Corollary 19. *Let* k_n *be a sequence with* $\lim_{n\to\infty}\frac{k_n}{n}=0$*, and let M denote the* k_n*-nearest neighbor algorithm. Then for any* $r>0$*, RobustNonPar*(M,r) *is r-consistent.*

Remark:

Corollary 19 gives a formal guarantee in the large sample limit for the modified nearest-neighbor algorithm proposed by [11].

Corollary 20. *Let W be a kernel classifier over* $\mathscr{X}\times\{\pm1\}$ *constructed from K and* h_n*. Suppose the following properties hold for K and* h_n*.*

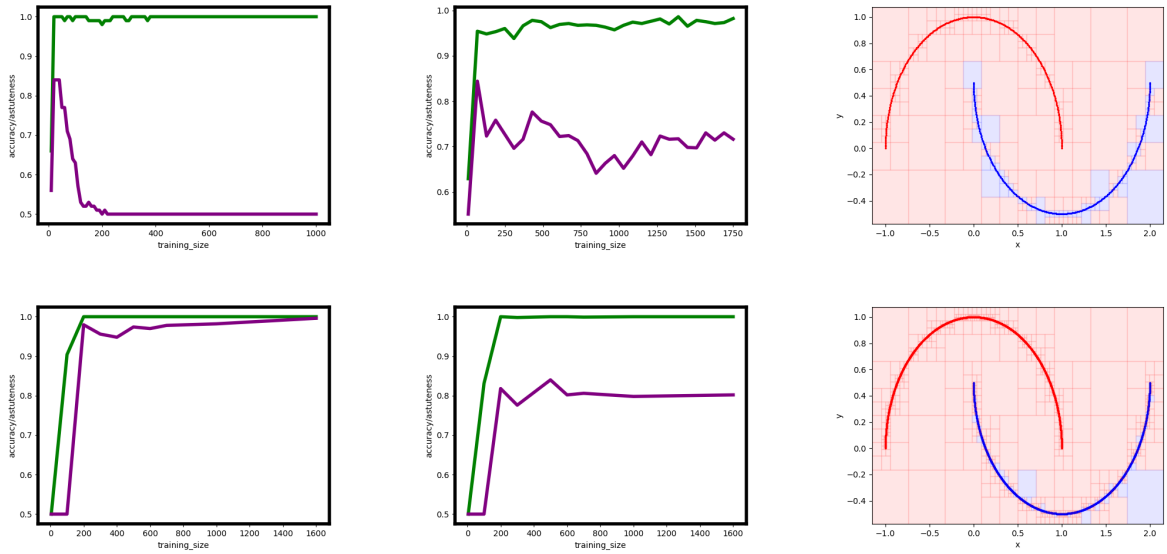


Figure 1.2. Empirical accuracy/astuteness of different classifiers as a function of training sample size. Accuracy is shown in green, astuteness in purple. Left : Noiseless Setting. Right: Noisy Setting. Top Row: Histogram Classifier, Bottom Row: 1-Nearest Neighbor

1. *For any* $c > 1$, $\lim_{x\to\infty} \frac{K(cx)}{K(x)} = 0$.

2. $\lim_{n\to\infty} h_n = 0$.

Then for any $r > 0$, $RobustNonPar(W, r)$ *is r-consistent.*

Observe again that Condition 1. is satisfied by any K that decreases more rapidly than an inverse polynomial kernel; it is thus satisfied by most popular kernels, such as the Gaussian kernel.

1.5 Validation

Our theoretical results are, by nature, large sample; we next validate how well they apply to the finite sample case by trying them out on a simple example. In particular, we ask the following question:

> How does the robustness of non-parametric classifiers change with increasing sample size?

This question is considered in the context of two simple non-parametric classifiers – one nearest neighbor (which is guaranteed to be r-consistent) and histograms (which is not). To be able to measure performance with increasing data size, we look at a simple synthetic dataset – the Half Moons.

1.5.1 Experimental Setup

Classifiers and Dataset.

We consider two different classification algorithms – one nearest neighbor (NN) and a Histogram Classifier (HC). We use the Halfmoon dataset with two settings of the gaussian noise parameter σ, $\sigma = 0$ (Noiseless) and $\sigma = 0.08$ (Noisy). For the Noiseless setting, observe that the data is already 0.1-separated; for the Noisy setting, we use Adversarial Pruning (Algorithm 1) with parameter $r = 0.1$ for both classification methods.

Performance Measure.

We evaluate robustness with respect to the ℓ_∞ metric, that is commonly used in the adversarial examples literature. Specifically, for each classifier, we calculate the *empirical astuteness*, which is the fraction of test examples on which it is astute.

Observe that computing the empirical astuteness of a classifier around an input x amounts to finding the adversarial example that is *closest to* x according to the ℓ_∞ norm. For the 1-nearest neighbor, we do this using the optimal attack algorithm proposed by Yang et. al. [11]. For the histogram classifier, we use the optimal attack framework proposed by [11], and show that the structure of the classifier can be exploited to solve the convex program efficiently. Details are in Appendix C.

We use an attack radius of $r = 0.1$ for the Noiseless setting, and $r = 0.09$ for the Noisy setting. For all classification algorithms, we plot the empirical astuteness as a function of the training set size. As a baseline, we also plot their standard accuracy on the test set.

1.5.2 Results

The results are presented in Figure 1.2; the left two panels are for the Noiseless setting while the two center ones are for the Noisy setting.

The results show that as predicted by our theory, for the Noiseless setting, the empirical astuteness of nearest neighbors converges to 1 as the training set grows. For Histogram Classifiers, the astuteness converges to 0.5 – indicating that the classifier may grow less and less astute with higher sample size even for well-separated data. This is plausibly because the cell size induced by the histogram grows smaller with growing training data; thus, the classifier that outputs the default label -1 in empty cells is incorrect on adversarial examples that are close to a point with $+1$ label, but belongs to a different, empty cell. The rightmost panels in Figure 1.2 provide a visual illustration of this process.

For the Noisy setting, the empirical astuteness of adversarial pruning followed by nearest neighbors converges to 0.8. For histograms with adversarial pruning, the astuteness converges to 0.7, which is higher than the noiseless case but still clearly sub-optimal.

1.5.3 Discussion

Our results show that even though our theory is asymptotic, our predictions continue to be relevant in finite sample regimes. In particular, on well-separated data, nearest neighbors that we theoretically predict to be intrinsically robust is robust; histogram classifiers, which do not satisfy the conditions in Theorem 12 are not. Our predictions continue to hold for data that is not well-separated. Nearest neighbors coupled with Adversarial Pruning continues to be robust with growing sample size, while histograms continue to be non-robust. Thus our theory is confirmed by practice.

1.6 Conclusion

In conclusion, we rigorously analyze when non-parametric methods provide classifiers that are robust in the large sample limit. We provide a general condition that characterizes when non-parametric methods are robust on well-separated data, and show that Adversarial Pruning of [11] works on data that is not well-separated.

Our results serve to provide a set of guidelines that can be used for designing non-parametric methods that are robust and accurate on well-separated data; additionally, we demonstrate that when data is not well-separated, preprocessing by adversarial pruning [11] does lead to optimally astute solutions in the large sample limit.

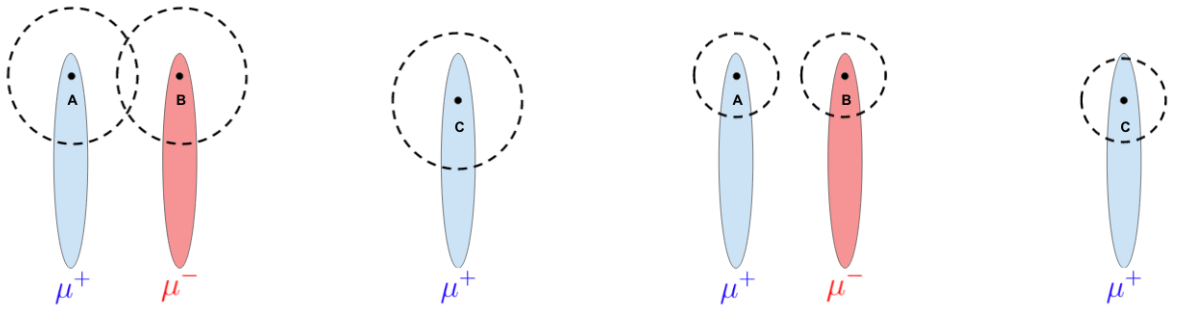
Chapter 2

Consistent Non-Parametric Methods for Maximizing Robustness

2.1 Introduction

Adversarially robust classification, that has been of much recent interest, is typically formulated as follows. We are given data drawn from an underlying distribution D, a metric d, as well as a pre-specified robustness radius r. We say that a classifier c is r-robust at an input x if it predicts the same label on a ball of radius r around x. Our goal in robust classification is to find a classifier c that maximizes astuteness, which is defined as accuracy on those examples where c is also r-robust.

While this formulation has inspired a great deal of recent work, both theoretical and empirical [12, 13, 14, 15, 1, 16, 18, 19, 20, 21, 34], a major limitation is that enforcing a pre-specified robustness radius r may lead to sub-optimal accuracy *and* robustness. To see this, consider what would be an ideally robust classifier the example in Figure 2.1. For simplicity, suppose that we know the data distribution. In this case, a classifier that has an uniformly large robustness radius r will misclassify some points from the blue cluster on the left, leading to lower accuracy. This is illustrated in panel (a), in which large robustness radius leads to intersecting robustness regions. On the other hand, in panel (b), the blue cluster on the right is highly separated from the red cluster, and could be accurately classified with a high margin. But this will not happen if the robustness radius is set small enough to avoid the problems posed in



(a) Large robustness radii

(b) Small robustness radii

Figure 2.1. A data distribution demonstrating the difficulties with fixed radius balls for robustness regions. The red represents negatively labeled points, and the blue positive. If the robustness radius is set too large (panel (a)), then the regions of A and B intersect leading to a loss of accuracy. If the radius is set too small (panel (b)), this leads to a loss of robustness at point C where in principle it should be possible to defend against a larger amount of adversarial attacks.

panel (a). Thus, enforcing a fixed robustness radius that applies to the entire dataset may lead to lower accuracy and lower robustness.

In this work, we propose an alternative formulation of robust classification that ensures that in the large sample limit, there is no robustness-accuracy trade off, and that regions of space with higher separation are classified more robustly. An extra advantage is that our formulation is achievable by existing methods. In particular, we show that two very common non-parametric algorithms – nearest neighbors and kernel classifiers – achieve these properties in the large sample limit.

Our formulation is built on the notion of a new large-sample limit. In the standard statistical learning framework, the large-sample ideal is the Bayes optimal classifier that maximizes accuracy on the data distribution, and is undefined outside. Since this is not always robust with radius r, prior work introduces the notion of an r-optimal classifier [11] that maximizes accuracy on points where it is also r-robust. However, this classifier also suffers from the same challenges as the example in Figure 2.1.

We depart from both by introducing a new limit that we call the neighborhood preserving Bayes optimal classifier, described as follows. Given an input x that lies in the support of the data distribution D, it predicts the same label as the Bayes optimal. On an x outside the support,

it outputs the prediction of the Bayes Optimal on the nearest neighbor of x *within* the support of D. The first property ensures that there is no loss of accuracy – since it always agrees with the Bayes Optimal within the data distribution. The second ensures higher robustness in regions that are better separated. Our goal is now to design classifiers that converge to the neighborhood preserving Bayes optimal in the large sample limit; this ensures that with enough data, the classifier will have accuracy approaching that of the Bayes optimal, as well as higher robustness where possible without sacrificing accuracy.

We next investigate how to design classifiers with this convergence property. Our starting point is classical statistical theory [10] that shows that a class of methods known as weight functions will converge to a Bayes optimal in the large sample limit provided certain conditions hold; these include k-nearest neighbors under certain conditions on k and n, certain kinds of decision trees as well as kernel classifiers. Through an analysis of weight functions, we next establish precise conditions under which they converge to the neighborhood preserving Bayes optimal in the large sample limit. As expected, these are stronger than standard convergence to the Bayes optimal. In the large sample limit, we show that k_n-nearest neighbors converge to the neighborhood preserving Bayes optimal provided $k_n = \omega(\log n)$, and kernel classifiers converge to the neighborhood preserving Bayes optimal provided certain technical conditions (such as the bandwidth shrinking sufficiently slowly). By contrast, certain types of histograms do not converge to the neighborhood preserving Bayes optimal, even if they do converge to the Bayes optimal. We round these off with a lower bound that shows that for nearest neighbor, the condition that $k_n = \omega(\log n)$ is tight. In particular, for $k_n = O(\log n)$, there exist distributions for which k_n-nearest neighbors provably fails to converge towards the neighborhood preserving Bayes optimal (despite converging towards the standard Bayes optimal).

In summary, the contributions of the paper are as follows. First, we propose a new large sample limit the neighborhood preserving Bayes optimal and a new formulation for robust classification. We then establish conditions under which weight functions, a class of non-parametric methods, converge to the neighborhood preserving Bayes optimal in the large sample

limit. Using these conditions, we show that k_n-nearest neighbors satisfy these conditions when $k_n = \omega(\log n)$, and kernel classifiers satisfy these conditions provided the kernel function K has faster than polynomial decay, and the bandwidth parameter h_n decreases sufficiently slowly.

To complement these results, we also include negative examples of non-parametric classifiers that do not converge. We provide an example where histograms do not converge to the neighborhood preserving Bayes optimal with increasing n. We also show a lower bound for nearest neighbors, indicating that $k_n = \omega(\log n)$ is both necessary and sufficient for convergence towards the neighborhood preserving Bayes optimal.

Our results indicate that the neighborhood preserving Bayes optimal formulation shows promise and has some interesting theoretical properties. We leave open the question of coming up with other alternative formulations that can better balance both robustness and accuracy for all kinds of data distributions, as well as are achievable algorithmically. We believe that addressing this would greatly help address the challenges in adversarial robustness.

2.2 Preliminaries

We consider binary classification over $\mathbb{R}^d \times \{\pm 1\}$, and let ρ denote any distance metric on $\mathbb{R}^d$. We let μ denote the measure over $\mathbb{R}^d$ corresponding to the probability distribution over which instances $x \in \mathbb{R}^d$ are drawn. Each instance x is then labeled as $+1$ with probability $\eta(x)$ and -1 with probability $1 - \eta(x)$. Together, μ and η comprise our data distribution $\mathscr{D} = (\mu, \eta)$ over $\mathbb{R}^d \times \{\pm 1\}$.

For comparison to the robust case, for a classifier $f : \mathbb{R}^d \to \{\pm 1\}$ and a distribution $\mathscr{D}$ over $\mathbb{R}^d \times \{\pm 1\}$, it will be instructive to consider its **accuracy**, denoted $A(f, \mathscr{D})$, which is defined as the fraction of examples from $\mathscr{D}$ that f labels correctly. Accuracy is maximized by the **Bayes Optimal classifier**: which we denote by g. It can be shown that for any $x \in supp(\mu)$, $g(x) = 1$ if $\eta(x) \geq \frac{1}{2}$, and $g(x) = -1$ otherwise.

Our goal is to build classifiers $\mathbb{R}^d \to \{\pm 1\}$ that are both accurate and robust to small

perturbations. For any example x, perturbations to it are constrained to taking place in the **robustness region** of x, denoted U_x. We will let $\mathscr{U} = \{U_x : x \in \mathbb{R}^d\}$ denote the collections of all robustness regions.

We say that a classifier $f : \mathbb{R}^d \to \{\pm 1\}$ is **robust** at x if for all $x' \in U_x$, $f(x') = f(x)$. Combining robustness and accuracy, we say that classifier is **astute** at a point x if it is both accurate and robust. Formally, we have the following definition.

Definition 21. *A classifier $f : \mathbb{R}^d \to \{\pm 1\}$ is said to be **astute** at (x,y) with respect to robustness collection $\mathscr{U}$ if $f(x) = y$ and f is robust at x with respect to $\mathscr{U}$. If $\mathscr{D}$ is a data distribution over $\mathbb{R}^d \times \{\pm 1\}$, the **astuteness** of f over $\mathscr{D}$ with respect to $\mathscr{U}$, denoted $A_{\mathscr{U}}(f,\mathscr{D})$, is the fraction of examples $(x,y) \sim \mathscr{D}$ for which f is astute at (x,y) with respect to $\mathscr{U}$. Thus*

$$A_{\mathscr{U}}(f,\mathscr{D}) = P_{(x,y)\sim\mathscr{D}}[f(x') = y, \forall x' \in \mathscr{U}_x].$$

Non-parametric Classifiers

We now briefly review several kinds of non-parametric classifiers that we will consider throughout this paper. We begin with *weight functions*, which are a general class of non-parametric algorithms that encompass many classic algorithms, including nearest neighbors and kernel classifiers.

Weight functions are built from training sets, $S = \{(x_1,y_1),(x_2,y_2,),\ldots,(x_n,y_n)\}$ by assigning a function $w_i^S : \mathbb{R}^d \to [0,1]$ that essentially scores how relevant the training point (x_i,y_i) is to the example being classified. The functions w_i^S are allowed to depend on $x_1,\ldots,x_n$ but must be independent of the labels $y_1,\ldots,y_n$. Given these functions, a point x is classified by just checking whether $\sum y_i w_i^S(x) \geq 0$ or not. If it is nonnegative, we output $+1$ and otherwise -1. A complete description of weight functions is included in the appendix.

Next, we enumerate several common Non-parametric classifiers that can be construed as weight functions. Details can be found in the appendix.

Histogram classifiers partition the domain $\mathbb{R}^d$ into cells recursively by splitting cells

that contain a sufficiently large number of points x_i. This corresponds to a weight function in which $w_i^S(x) = \frac{1}{k_x}$ if x_i is in the same cell as x, where k_x denotes the number of points in the cell containing x.

k_n-**nearest neighbors** corresponds to a weight function in which $w_i^S(x) = \frac{1}{k_n}$ if x_i is one of the k_n nearest neighbors of x, and $w_i^S(x) = 0$ otherwise.

Kernel-Similarity classifiers are weight functions built from a kernel function $K : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ and a window size $(h_n)_1^\infty$ such that $w_i^S(x) \propto K(\rho(x,x_i)/h_n)$ (we normalize by dividing by $\sum_1^n K((\rho(x,x_i)/h_n)))$.

2.3 The Neighborhood preserving Bayes optimal classifier

Robust classification is typically studied by setting the robustness regions, $\mathscr{U} = \{U_x\}_{x\in\mathbb{R}^d}$, to be balls of radius r centered at x, $U_x = \{x' : \rho(x,x') \leq r\}$. The quantity r is the robustness radius, and is typically set by the practitioner (before any training has occurred).

This method has a limitation with regards to trade-offs between accuracy and robustness. To increase the margin or robustness, we must have a large robustness radius (thus allowing us to defend from larger adversarial attacks). However, with large robustness radii, this can come at a cost of accuracy, as it is not possible to robustly give different labels to points with intersecting robustness regions.

For an illustration, consider Figure 2.1. Here we consider a data distribution $D = (\mu, \eta)$ in which the blue regions denote all points with $\eta(x) > 0.5$ (and thus should be labeled $+$), and the red regions denote all points with $\eta(x) < 0.5$ (and thus should be labeled $-$). Observe that it is not possible to be simultaneously accurate and robust at points A, B while enforcing a large robustness radius, as demonstrated by the intersecting balls. While this can be resolved by using a smaller radius, this results in losing out on potential robustness at point C. In principal, we should be able to afford a large margin of robustness about C due to its relatively far distance from the red regions.

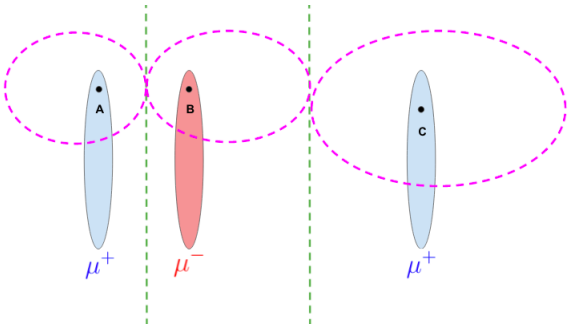


Figure 2.2. The decision boundary of the neighborhood preserving Bayes optimal classifier is shown in green, and the neighborhood preserving robust region of x is shown in pink. The former consists of points equidistant from μ^+, μ^-, and the latter consists of points equidistant from x, μ^+.

Motivated by this issue, we seek to find a formalism for robustness that allows us to simultaneously avoid paying for any accuracy-robustness trade-offs and *adaptively* size robustness regions (thus allowing us to defend against a larger range of adversarial attacks at points that are located in more homogenous zones of the distribution support). To approach this, we will first provide an ideal limit object: a classifier that has the same accuracy as the Bayes optimal (thus meeting our first criteria) that has good robustness properties. We call this the the neighborhood preserving Bayes optimal classifier, defined as follows.

Definition 22. *Let $\mathscr{D} = (\mu, \eta)$ be a distribution over $\mathbb{R}^d \times \{\pm 1\}$. Then the **neighborhood preserving Bayes optimal classifier of** $\mathscr{D}$, denoted $g_{neighbor}$, is the classifier defined as follows. Let $\mu^+ = \{x : \eta(x) \geq \frac{1}{2}\}$ and $\mu^- = \{x : \eta(x) < \frac{1}{2}\}$. Then for any $x \in \mathbb{R}^d$, $g_{neighbor}(x) = +1$ if $\rho(x, \mu^+) \leq \rho(x, \mu^-)$, and $g_{neighbor}(x) = -1$ otherwise.*

This classifier can be thought of as the most robust classifier that matches the accuracy of the Bayes optimal. We call it *neighborhood preserving* because it extends the Bayes optimal classifier into a local neighborhood about every point in the support. For an illustration, refer to Figure 2.2, which plots the decision boundary of the neighborhood preserving Bayes optimal for an example distribution.

Next, we turn our attention towards measuring its robustness, which must be done with respect to some set of robustness regions $\mathscr{U} = \{U_x\}$. While these regions U_x can be nearly

arbitrary, we seek regions U_x such that $A_{\mathscr{U}}(g_{max}, \mathscr{D}) = A(g_{bayes}, \mathscr{D})$ (our astuteness equals the maximum possible accuracy) and U_x are "as large as possible" (representing large robustness). To this end, we propose the following regions.

Definition 23. *Let $\mathscr{D} = (\mu, \eta)$ be a data distribution over $\mathbb{R}^d \times \{\pm 1\}$. Let $\mu^+ = \{x : \eta(x) > \frac{1}{2}\}$, $\mu^- = \{x : \eta(x) < \frac{1}{2}\}$, and $\mu^{1/2} = \{x : \eta(x) = \frac{1}{2}\}$. For $x \in \mu^+$, we define the **neighborhood preserving robustness region**, denoted V_x, as*

$$V_x = \{x' : \rho(x, x') < \rho(\mu^- \cup \mu^{\frac{1}{2}}, x')\}.$$

It consists of all points that are closer to x than they are to $\mu^- \cup \mu^{1/2}$ (points oppositely labeled from x). We can use a similar definition for $x \in \mu^-$. Finally, if $x \in \mu^{1/2}$, we simply set $V_x = \{x\}$.

These robustness regions take advantage of the structure of the neighborhood preserving Bayes optimal. They can essentially be thought of as regions that maximally extend from any point x in the support of $\mathscr{D}$ to the decision boundary of the neighborhood preserving Bayes optimal. We include an illustration of the regions V_x for an example distribution in Figure 2.2.

As a technical note, for $x \in supp(\mathscr{D})$ with $\eta(x) = 0.5$, we give them a trivial robustness region. The rational for doing this is that $\eta(x) = 0.5$ is an edge case that is arbitrary to classify, and consequently enforcing a robustness region at that point is arbitrary and difficult to enforce.

We now formalize the robustness and accuracy guarantees of the max-margin Bayes optimal classifier with the following two results.

Theorem 24. *(Accuracy) Let $\mathscr{D}$ be a data distribution. Let $\mathscr{V}$ denote the collection of neighborhood preserving robustness regions, and let g denote the Bayes optimal classifier. Then the neighborhood preserving Bayes optimal classifier, $g_{neighbor}$, satisfies $A_{\mathscr{V}}(g_{neighbor}, \mathscr{D}) = A(g, \mathscr{D})$, where $A(g, \mathscr{D})$ denotes the accuracy of the Bayes optimal. Thus, $g_{neighbor}$ maximizes accuracy.*

Theorem 25. *(Robustness) Let $\mathscr{D}$ be a data distribution, let f be a classifier, and let $\mathscr{U}$ be a set of robustness regions. Suppose that $A_{\mathscr{U}}(f, \mathscr{D}) = A(g, \mathscr{D})$, where g denotes the Bayes optimal*

classifier. Then there exists $x \in supp(\mathscr{D})$ such that $V_x \not\subset U_x$, where V_x denotes the neighborhood preserving robustness region about x. In particular, we cannot have V_x be a strict subset of U_x for all x.

Theorem 24 shows that the neighborhood preserving Bayes classifier achieves maximal accuracy, while Theorem 25 shows that achieving a strictly higher robustness (while maintaining accuracy) is not possible; while it is possible to make accurate classifiers which have higher robustness than $g_{neighbor}$ in some regions of space, it is not possible for this to hold across all regions. Thus, the neighborhood preserving Bayes optimal classifier can be thought of as a local maximum to the constrained optimization problem of maximizing robustness subject to having maximum (equal to the Bayes optimal) accuracy.

2.3.1 Neighborhood Consistency

Having defined the neighborhood preserving Bayes optimal classifier, we now turn our attention towards building classifiers that converge towards it. Before doing this, we must precisely define what it means to converge. Intuitively, this consists of building classifiers whose robustness regions "approach" the robustness regions of the neighborhood preserving Bayes optimal classifier. This motivates the definition of *partial neighborhood preserving robustness regions*.

Definition 26. *Let $0 < \kappa < 1$ be a real number, and let $\mathscr{D} = (\mu, \eta)$ be a data distribution over $\mathbb{R}^d \times \{\pm 1\}$. Let $\mu^+ = \{x : \eta(x) > \frac{1}{2}\}$, $\mu^- = \{x : \eta(x) < \frac{1}{2}\}$, and $\mu^{1/2} = \{x : \eta(x) = \frac{1}{2}\}$. For $x \in \mu^+$, we define the **neighborhood preserving robustness region**, denoted V_x, as*

$$V_x = \{x' : \rho(x,x') < \kappa\rho(\mu^- \cup \mu^{\frac{1}{2}}, x')\}.$$

It consists of all points that are closer to x than they are to $\mu^- \cup \mu^{1/2}$ (points oppositely labeled from x) by a factor of κ. We can use a similar definition for $x \in \mu^-$. Finally, if $\eta(x) = \frac{1}{2}$, we simply set $V_x^\kappa = \{x\}$.

Observe that $V_x^\kappa \subset V_x$ for all $0 < \kappa < 1$, and thus being robust with respect to V_x^κ is a milder condition than V_x. Using this notion, we can now define margin consistency.

Definition 27. *A learning algorithm A is said to be* ***neighborhood consistent*** *if the following holds for any data distribution $\mathscr{D} = (\mu, \eta)$ where η is continuous on its support. For any $0 < \varepsilon, \delta, \kappa < 1$, there exists N such that for all $n \geq N$, with probability at least $1 - \delta$ over $S \sim \mathscr{D}^n$,*

$$A_{\mathscr{V}^\kappa}(A_S, D) \geq A(g, \mathscr{D}) - \varepsilon,$$

where g denotes the Bayes optimal classifier and A_S denotes the classifier learned by algorithm A from dataset S.

This condition essentially says that the astuteness of the classifier learned by the algorithm converges towards the accuracy of the Bayes optimal classifier. Furthermore, we stipulate that this holds as long as the astuteness is measured with respect to some $\mathscr{V}^\kappa$. Observe that as $\kappa \to 1$, these regions converge towards the neighborhood preserving robustness regions, thus giving us a classifier with robustness effectively equal to that of the neighborhood preserving Bayes optimal classifier.

2.4 Neighborhood Consistent Non-Parametric Classifiers

Having defined neighborhood consistency, we turn to the following question: which non-parametric algorithms are neighborhood consistent? Our starting point will be the standard literature for the convergence of non-parametric classifiers with regard to accuracy. We begin by considering the standard conditions for k_n-nearest neighbors to converge (in accuracy) towards the Bayes optimal.

k_n-nearest neighbors is *consistent* if and only if the following two conditions are met: $\lim_{n\to\infty} k_n = \infty$, and $\lim_{n\to\infty} \frac{k_n}{n} = 0$. The first condition guarantees that each point is classified by using an increasing number of nearest neighbors (thus making the probability of a misclassification small), and the second condition guarantees that each point is classified using only points

very close to it. We will refer to the first condition as *precision*, and the second condition as *locality.* A natural question is whether the same principles suffice for neighborhood consistency as well. We began by showing that without any additional constraints, the answer is no.

Theorem 28. *Let $\mathscr{D} = (\mu, \eta)$ be the data distribution where μ denotes the uniform distribution over $[0,1]$ and η is defined as: $\eta(x) = x$. Over this space, let ρ be the euclidean distance metric. Suppose $k_n = O(\log n)$ for $1 \leq n < \infty$. Then k_n-nearest neighbors is not neighborhood consistent with respect to $\mathscr{D}$.*

The issue in the example above is that for smaller k_n, k_n-nearest neighbors lacks sufficient precision. For neighborhood consistnecy, points must be labeled using even more training points than are needed accuracy. This is because the classifier must be uniformly correct across the entirety of V_x^κ. Thus, to build neighborhood consistent classifiers, we must bolster the precision from the standard amount used for standard consistency. To do this, we begin by introducing *splitting numbers*, a useful tool for bolstering the precision of weight functions.

2.4.1 Splitting Numbers

We will now generalize beyond nearest neighbors to consider weight functions. Doing so will allow us to simultaneously analyze nearest neighbors and kernel classifiers. To do so, we must first rigorously substantiate our intuitions about increasing precision into concrete requirements. This will require several technical definitions.

Definition 29. *Let μ be a probability measure over $\mathbb{R}^d$. For any $x \in \mathbb{R}^d$, the **probability radius** $r_p(x)$ is the smallest radius for which $B(x, r_p(x))$ has probability mass at least p. More precisely, $r_p(x) = \inf\{r : \mu(B(x,r)) \geq p\}$.*

Definition 30. *Let W be a weight function and let $S = \{x_1, x_2, \ldots, x_n\}$ be any finite subset of $\mathbb{R}^d$. For any $x \in \mathbb{R}^d$, $\alpha \geq 0$, and $0 \leq \beta \leq 1$, let $W_{x,\alpha,\beta} = \{i : \rho(x,x_i) \leq \alpha, w_i^S(x) \geq \beta\}$. Then the **splitting number** of W with respect to S, denoted as $T(W,S)$ is the number of distinct subsets*

generated by $W_{x,\alpha\beta}$ as x ranges over $\mathbb{R}^d$, α ranges over $[0,\infty)$, and β ranges over $[0,1]$. Thus $T(W,S) = |\{W_{x,\alpha,\beta} : x \in \mathbb{R}^d, 0 \leq \alpha, 0 \leq \beta \leq 1\}|$.

Splitting numbers allow us to ensure high amounts of precision over a weight function. To prove neighborhood consistency, it is necessary for a classifier to be correct at *all* points in a given region. Consequently, techniques that consider a single point will be insufficient. The splitting number provides a mechanism for studying entire regions simultaneously. For more details on splitting numbers, we include several examples in the appendix.

2.4.2 Sufficient Conditions for Neighborhood Consistency

We now state our main result.

Theorem 31. *Let W be a weight function, $\mathscr{D}$ a distribution over $\mathbb{R}^d \times \{\pm 1\}$, $\mathscr{U}$ a neighborhood preserving collection, and $(t_n)_1^\infty$ be a sequence of positive integers such that the following four conditions hold.*

1. W is consistent (with resp. to accuracy) with resp. to $\mathscr{D}$.

2. For any $0 < p < 1$, $\lim_{n\to\infty} E_{S\sim\mathscr{D}^n}[\sup_{x\in\mathbb{R}^d} \sum_1^n w_i^S(x) 1_{\rho(x,x_i)>r_p(x)}] = 0$.

3. $\lim_{n\to\infty} E_{S\sim D^n}[t_n \sup_{x\in\mathbb{R}^d} w_i^S(x)] = 0$.

4. $\lim_{n\to\infty} E_{S\sim D^n} \frac{\log T(W,S)}{t_n} = 0$.

Then W is neighborhood consistent with respect to $\mathscr{D}$.

Remarks: Condition 1 is necessary because neighborhood consistency implies standard consistency – or, convergence in accuracy to the Bayes Optimal. Standard consistency has been well studied for non-parametric classifiers, and there are a variety of results that can be used to ensure it – for example, Stone's Theorem (included in the appendix).

Conditions 2. and 3. are stronger version of conditions 2. and 3. of Stone's theorem. In particular, both include a supremum taken over all $x \in \mathbb{R}^d$ as opposed to simply considering a random point $x \sim \mathscr{D}$. This is necessary for ensuring correct labels on entire regions of points simultaneously. We also note that the dependence on $r_p(x)$ (as opposed to some fixed r) is a key

property used for adaptive robustness. This allows the algorithm to adjust to potential differing distance scales over different regions in $\mathbb{R}^d$. This idea is reminiscent of the analysis given in [35], which also considers probability radii.

Condition 4. is an entirely new condition which allows us to simultaneously consider all $T(W,S)$ subsets of S. This is needed for analyzing weighted sums with arbitrary weights.

Next, we apply Theorem 31 to get specific examples of margin consistent non-parametric algorithms.

2.4.3 Nearest Neighbors and Kernel Classifiers

We now provide sufficient conditions for k_n-nearest neighbors to be neighborhood consistent.

Corollary 32. *Suppose* $(k_n)_1^\infty$ *satisfies (1)* $\lim_{n\to\infty} \frac{k_n}{n} = 0$*, and (2)* $\lim_{n\to\infty} \frac{\log n}{k_n} = 0$*. Then* k_n*-nearest neighbors is neighborhood consistent.*

As a result of Theorem 28, corollary 32 is tight for nearest neighbors. Thus k_n nearest neighbors is neighborhood consistent if and only if $k_n = \omega(\log n)$.

Next, we give sufficient conditions for a kernel-similarity classifier.

Corollary 33. *Let* W *be a kernel classifier over* $\mathbb{R}^d \times \{\pm 1\}$ *constructed from* $K : \mathbb{R}^+ \to \mathbb{R}^+$ *and* h_n*. Suppose the following properties hold.*

1. K *is decreasing, and satisfies* $\int_{\mathbb{R}^d} K(||x||)dx < \infty$.

2. $\lim_{n\to\infty} h_n = 0$ *and* $\lim_{n\to\infty} nh_n^d = \infty$.

3. For any $c > 1$, $\lim_{x\to\infty} \frac{K(cx)}{K(x)} = 0$.

4. For any $x \geq 0$, $\lim_{n\to\infty} \frac{n}{\log n} K(\frac{x}{h_n}) = \infty$.

Then W *is neighborhood consistent.*

Observe that conditions 1. 2. and 3. are satisfied by many common Kernel functions such as the Gaussian or Exponential kernel ($K(x) = \exp(-x^2)$/ $K(x) = \exp(-x)$). Condition 4. can be

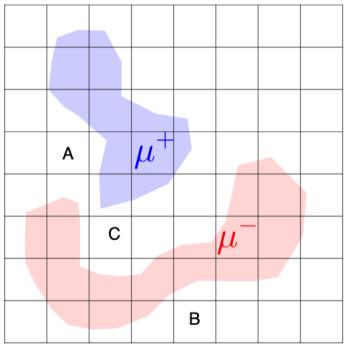


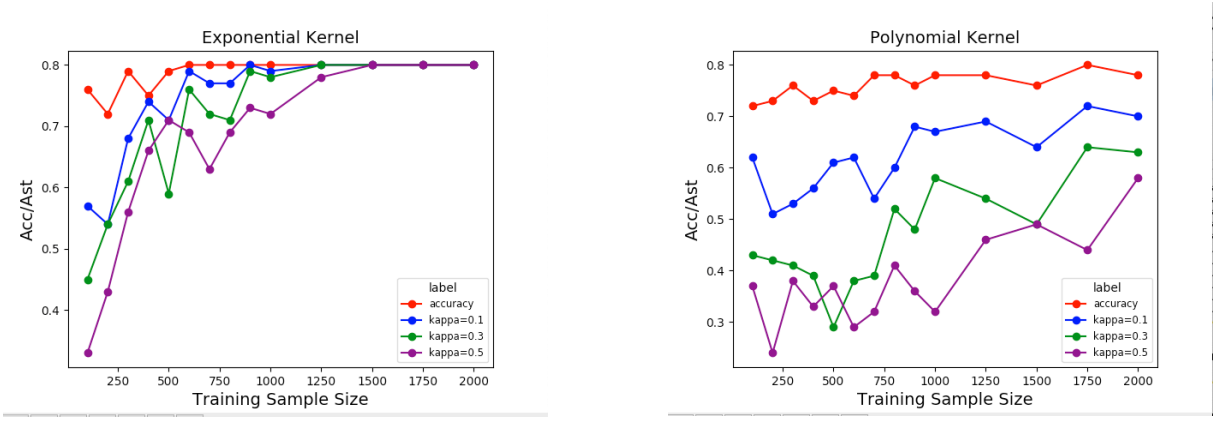
Figure 2.3. we have a histogram classifier being applied to the blue and red regions. The classifier will be unable to construct good labels in the cells labeled A, B, C, and consequently will not be robust with respect to V_x^κ for sufficiently large κ.

similarly satisfied by just increasing h_n to be sufficiently large. Overall, this theorem states that Kernel classification is neighborhood consistent as long as the bandwidth shrinks slowly enough.

2.4.4 Histogram Classifiers

Having discussed neighborhood consistent nearest-neighbors and kernel classifier, we now turn our attention towards another popular weight function, histogram classifiers. Recall that histogram classifiers operate by partitioning their input space into increasingly small cells, and then classifying each cell by using a majority vote from the training examples within that cell (a detailed description can be found in the appendix). We seek to answer the following question: is increasing precision sufficient for making histogram classifiers neighborhood consistent? Unfortunately, the answer this turns out not to be no. The main issue is that histogram classifiers have no mechanism for performing classification outside the support of the data distribution.

For an example of this, refer to Figure 2.3. Here we see a distribution being classified by a histogram classifier. Observe that the cell labeled A contains points that are strictly closer to μ^+ than μ^-, and consequently, for sufficiently large κ, V_x^κ will intersect A for some point $x \in \mu^+$. A similar argument holds for the cells labeled B and C.. However, since A, B, C are all in cells that will never contain any data, they will never be labeled in a meaningful way. Because of this, histogram classifiers are not neighborhood consistent.



(a) exponential kernel

(b) polynomial kernel

Figure 2.4. Plots of astuteness against the training sample size. In both panels, accuracy is plotted in red, and the varying levels of robustness regions ($\kappa = 0.1, 0.3, 0.5$) are givne in blue, green and purple. In panel (a), observe that as sample size increases, every measure of astuteness converges towards 0.8 which is as predicted by Corollary 33. In panel (b), although the accuracy appears to converge, none of the robustness measure. In fact, they get progressively worse the larger κ gets.

2.5 Validation

To complement our theoretical large sample results for non-parametric classifiers, we now include several experiments to understand their behavior for finite samples. We seek to understand how quickly non-parametic classifiers converge towards the neighborhood preserving Bayes optimal.

We focus our attention on kernel classifiers and use two different kernel similarity functions: the first, an exponential kernel, and the second, a polynomial kernel. These classifiers were chosen so that the former meets the conditions of Corollary 33, and the latter does not. Full details on these classifiers can be found in the appendix.

To be able to measure performance with increasing data size, we look at a simple synthetic dataset over overlayed circles (see Figure A.1 for an illustration) with support designed so that the data is intrinsically multiscaled. In particular, this calls for different levels of robustness in different regions. For simplicity, we use a global label noise parameter of 0.2, meaning that any sample drawn from this distribution is labeled differently than its support with probability 0.2.

Further details about our dataset are given in section A.4.

Performance Measure. For a given classifier, we evaluate its astuteness at a test point x with respect to the robustness region V_x^κ (Definition 26). While these regions are not computable in practice due to their dependency on the support of the data distribution, we are able to approximate them for this synthetic example due to our explicit knowledge of the data distribution. Details for doing this can be found in the appendix. To compute the empirical astuteness of a kernel classifier W_K about test point x, we perform a grid search over all points in V_x^κ to ensure that all points in the robustness region are labeled correctly.

For each classifier, we measure the empirical astuteness by using three trials of 20 test points and taking the average. While this is a relatively small amount of test data, it suffices as our purpose is to just verify that the algorithm roughly converges towards the optimal possible astuteness. Recall that for any neighborhood consistent algorithm, as $n \to \infty$, $A_{\gamma\kappa}$ should converge towards A^*, the accuracy of the Bayes optimal classifier, for *any* $0 < \kappa < 1$. Thus, to verify this holds, we use $\kappa = 0.1, 0.3, 0.5$. For each of these values, we plot the empirical astuteness as the training sample size n gets larger and larger. As a baseline, we also plot their standard accuracy on the test set.

Results and Discussion: The results are presented in Figure 2.4; the left panel is for the exponential kernel, while the right one is for the polynomial kernel. As predicted by our theory, we see that in all cases, the exponential kernel converges towards the maximum astuteness regardless of the value of κ: the only difference is that the rate of convergence is slower for larger values of κ. This is, of course, expected because larger values of κ entail larger robustness regions.

By contrast, the polynomial kernel performs progressively worse for larger values of κ. This kernel was selected specifically to violate the conditions of Corollary 33, and in particular fails criteria 3. However, note that the polynomial kernel nevertheless performs will with respect to accuracy thus giving another example demonstrating the added difficulty of neighborhood consistency.

Our results bridge the gap between our asymptotic theoretical results and finite sample regimes. In particular, we see that kernel classifiers that meet the conditions of Corollary 33 are able to converge in astuteness towards the neighborhood preserving Bayes optimal classifier, while classifiers that do not meet these conditions fail.

2.6 Related Work

There is a wealth of literature on robust classification, most of which impose the same robustness radius r on the entire data. [12, 13, 14, 15, 1, 16, 17, 18, 19, 20, 21], among others, focus primarily on neural networks, and robustness regions that are ℓ_1, ℓ_2, or ℓ_∞ norm balls of a given radius r.

[36] and [37] show how to train neural networks with different robustness radii at different points by trading off robustness and accuracy; their work differ from ours in that they focus on neural networks, their robustness regions are still norm balls, and that their work is largely empirical.

Our framework is also related to large margin classification – in the sense that the robustness regions $\mathscr{U}$ induce a *margin constraint* on the decision boundary. The most popular large margin classifier is the Support Vector Machine[38, 39, 40] – a large margin linear classifier that minimizes the worst-case margin over the training data. Similar ideas have also been used to design classifiers that are more flexible than linear; for example, [41] shows how to build large margin Lipschitz classifiers by rounding globally Lipschitz functions. Finally, there has also been purely empirical work on achieving large margins for more complex classifiers – such as [42] for deep neural networks that minimizes the worst case margin, and [43] for metric learning to find large margin nearest neighbors. Our work differs from these in that our goal is to ensure a high enough local margin at each x, (by considering the neighborhood preserving regions V_x) as opposed to optimizing a global margin.

Finally, our analysis builds on prior work on robust classification for non-parametric

methods in the standard framework. [22, 23, 5, 11] provide adversarial attacks on non-parametric methods. Wang et. al. [5] develops a defense for 1-NN that removes a subset of the training set to ensure higher robustness. Yang et. al [11] proposes the r-optimal classifier – which is the maximally astute classifier in the standard robustness framework – and proposes a defense called Adversarial Pruning.

Theoretically, [44] provide conditions under which weight functions converge towards the r-optimal classifier in the large sample limit. They show that for r-separated distributions, where points from different classes are at least distance $2r$ or more apart, nearest neighbors and kernel classifiers satisfy these conditions. In the more general case, they use Adversarial Pruning as a preprocessing step to ensure that the training data is r-separated, and show that this preprocessing step followed by nearest neighbors or kernel classifiers leads to solutions that are robust and accurate in the large sample limit. Our result fundamentally differs from theirs in that we analyze a different algorithm, and our proof techniques are quite different. In particular, the fundamental differences between the r-optimal classifier and the neighborhood preserving Bayes optimal classifier call for different algorithms and different analysis techniques.

In concurrent work, [45] proposes a similar limit to the neighborhood preserving Bayes optimal which they refer to as the margin canonical Bayes. However, their work then focuses on a data augmentation technique that leads to convergence whereas we focus on proving the neighborhood consistency of classical non-parametric classifiers.

Appendix A

Appendix for Chapter 2

A.1 Further Details of Definitions and Theorems

A.1.1 Non-Parametric Classifiers

In this section, we precisely define weight functions, histogram classifiers and kernel classifiers.

Definition 34. *[9] A **weight function** W is a non-parametric classifier with the following properties.*

1. *Given input $S = \{(x_1, y_1), (x_2, y_2,), \ldots, (x_n, y_n)\} \sim \mathscr{D}^n$, W constructs functions $w_1^S, w_2^S, \ldots, w_n^S$: $\mathbb{R}^d \to [0,1]$ such that for all $x \in \mathbb{R}^d$, $\sum_1^n w_i^S(x) = 1$. The functions w_i^S are allowed to depend on $x_1, x_2, \ldots x_n$ but must be independent of $y_1, y_2, \ldots, y_n$.*

2. *W has output W_S defined as*

$$W_S(x) = \begin{cases} +1 & \sum_1^n w_i^S(x) y_i > 0 \\ -1 & \sum_1^n w_i^S(x) y_i \leq 0 \end{cases}$$

As a result, $w_i^S(x)$ can be thought of as the weight that (x_i, y_i) has in classifying x.

Definition 35. *A **histogram classifier**, H, is a non-parametric classification algorithm over $\mathbb{R}^d \times \{\pm 1\}$ that works as follows. For a distribution $\mathscr{D}$ over $\mathbb{R} \times \{\pm 1\}$, H takes $S = \{(x_i, y_i) :$*

$1 \leq i \leq n\} \sim \mathscr{D}^n$ *as input. Let* k_i *be a sequence with* $\lim_{i\to\infty} k_i = \infty$ *and* $\lim_{i\to\infty} \frac{k_i}{i} = 0$. *H constructs a set of hypercubes* $C = \{c_1, c_2, \ldots, c_m\}$ *as follows:*

1. *Initially* $C = \{c\}$, *where* $S \subset c$.
2. *For* $c \in C$, *if c contains more than* k_n *points of S, then partition c into* 2^d *equally sized hypercubes, and insert them into C.*
3. *Repeat step* 2 *until all cubes in C have at most* k_n *points.*

For $x \in \mathbb{R}$ *let* $c(x)$ *denote the unique cell in C containing x. If* $c(x)$ *doesn't exist, then* $H_S(x) = -1$ *by default. Otherwise,*

$$H_S(x) = \begin{cases} +1 & \sum_{x_i \in c(x)} y_i > 0 \\ -1 & \sum_{x_i \in c(x)} y_i \leq 0 \end{cases}.$$

Definition 36. *A* ***partitioning rule*** *is a weight function W over* $\mathscr{X} \times \{\pm 1\}$ *constructed in the following manner. Given* $S = \{(x_i, y_i)\} \sim \mathscr{D}^n$, *as a function of* $\{x_1, \ldots, x_n\}$, *we partition* $\mathbb{R}^d$ *into regions with* $A(x)$ *denoting the region containing x. Then, for any* $x \in \mathbb{R}^d$ *we have*

$$w_i^S(x) = \begin{cases} 1 & x_i \in A(x) \\ 0 & otherwise \end{cases}.$$

To achieve $\sum w_i^S(x) = 1$, *we can simply normalize weights for any x by* $\sum_1^n w_i^S(X)$.

Definition 37. *A* ***kernel classifier*** *is a weight function W over* $\mathbb{R}^d \times \{\pm 1\}$ *constructed from function* $K : \mathbb{R}^+ \cup \{0\} \to \mathbb{R}^+$ *and some sequence* $\{h_n\} \subset \mathbb{R}^+$ *in the following manner. Given* $S = \{(x_i, y_i)\} \sim \mathscr{D}^n$, *we have*

$$w_i^S(x) = \frac{K(\frac{\rho(x,x_i)}{h_n})}{\sum_{j=1}^n K(\frac{\rho(x,x_j)}{h_n})}.$$

Then, as above, W has output

$$W_S(x) = \begin{cases} +1 & \sum_1^n w_i^S(x) y_i > 0 \\ -1 & \sum_1^n w_i^S(x) y_i \leq 0 \end{cases}$$

A.1.2 Splitting Numbers

We refer to definitions 29 and 30.

The main idea behind splitting numbers is that they allow us to ensure uniform convergence properties over a weight function. To prove neighborhood consistency, it is necessary for a classifier to be correct at *all* points in a given region. Consequently, techniques that consider a single point will be insufficient. The splitting number provides a mechanism for studying entire regions simultaneously. For clarity, we include a quick example in which we bound the splitting number for a given weight function.

Example:

Let W denote any kernel classifier corresponding such that $K : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is a decreasing function. For any $S \sim \mathscr{D}^n$, observe that the condition $w_i^S(x) \geq \beta$ precisely corresponds to $\rho(x, x_i) \leq \gamma$ for some value of γ. This is because $w_i^S(x) > w_j^S(x)$ if and only if $\rho(x, x_i) < \rho(x, x_j)$. Thus, the regions $W_{x,\alpha,\beta}$ correspond to $\{i : \rho(x, x_i) \leq \gamma\}$, where γ is a positive real number that depends on x, α, β. These sets precisely correspond to subsets of S that are contained within $B(x, \gamma)$. Since balls have VC dimension at most $d+2$, by Sauer's lemma, the number of subsets of S that can be obtained in this manner is $O(n^{d+2})$. Therefore, we have that $T(W, S) = O(n^{d+2})$ for all $S \sim \mathscr{D}^n$.

A.1.3 Stone's Theorem

Theorem 38. *[10] Let W be weight function over $\mathbb{R}^d \times \{\pm 1\}$. Suppose the following conditions hold for any distribution $\mathscr{D}$ over $\mathbb{R}^d \times \{\pm 1\}$. Let X be a random variable with distribution $\mathscr{D}_{\mathbb{R}^d}$,*

and $S = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\} \sim \mathscr{D}^n$. *All expectations are taken over* X *and* S.

1. There is a constant c *such that, for every nonnegative measurable function* f *satisfying* $\mathbb{E}[f(X)] < \infty$, *and* $\mathbb{E}[\sum_1^n w_i^S(X) f(x_i)] \leq c\mathbb{E}[f(x)]$.

2. $\forall a > 0$, $\lim_{n\to\infty} \mathbb{E}[\sum_1^n w_i^S(x) I_{||x_i - X|| > a||}] = 0$.

3. $\lim_{n\to\infty} \mathbb{E}[\max_{1\leq i\leq n} w_i^S(X)] = 0$.

Then W *is consistent.*

A.2 Proofs

Notation:

- We let ρ denote our distance metric over $\mathbb{R}^d$. For sets $X_1, X_2 \subset \mathbb{R}^d$, we let $\rho(X_1, X_2) = \inf_{x_1 \in X_1, x_2 \in X_2} \rho(x_1, x_2)$.
- For any $x \in \mathbb{R}^d$, $B(x,a) = \{x : \rho(x,x') \leq a\}$.
- For any measure over $\mathbb{R}^d$, μ, we let $supp(\mu) = \{x : \mu(B(x,a)) > 0 \text{ for all } a > 0\}$.
- Given some measure μ over $\mathbb{R}^d$ and some $x \in \mathbb{R}^d$, we let $r_p(x)$ denote the probability radius (Definition 29) of x with probability p. that is, $r_p(x) = \inf\{r : \mu(B(x,r)) \geq p\}$.
- For weight function W and training sample S, we let W_S denote the weight function learned by W from S.

A.2.1 Proofs of Theorems 24 and 25

Proof. (Theorem 24) Let $\mathscr{D} = (\mu, \eta)$ be a data distribution, and let μ^+, μ^- be as described in Definition 22. Observe that for any $x \in \mu^+$, the Bayes optimal classifier and the neighborhood preserving Bayes optimal both have the same output, and furthermore the neighborhood preserving Bayes gives this output (by definition) throughout the entirety of V_x, the neighborhood preserving robustness region of x. It follows that the neighborhood preserving Bayes optimal has optimal astuteness, as desired. □

Proof. (Theorem 25) Let $\mathscr{D} = (\mu, \eta)$ be a data distribution, and assume towards a contradiction that there exists classifier f which has maximal astuteness with respect towards some set of robustness regions $\mathscr{U} = \{U_x\}$ such that $V_x \subseteq U_x$ for all x. The key observation is that because f has maximal astuteness, we must have $f(x) = g(x)$ for almost all points $x \sim \mu$ (where g is the Bayes optimal classifier). Furthermore, for those values of x, we must have g be robust at x (meaning it uniformly outputs the same output through U_x).

In order for U_x to be strictly larger than V_x for some x, it *necessarily* must intersect with $U_{x'}$ for some x' with $g(x') \neq g(x)$, and this is what causes the contradiction: f cannot be astute at both x and x' if they are differently labeled and their robustness regions intersect. □

A.2.2 Proof of Theorem 28

Let $\mathscr{D} = (\mu, \eta)$ be the distribution with μ being the uniform distribution over $[0,1]$ and $\eta : [0,1] \to [0,1]$ be $\eta(x) = x$. For example, if $(x,y) \sim \mathscr{D}$, then $\Pr[y = 1 | x = 0.3] = 0.3$.

We desire to show that k_n-nearest neighbors is not neighborhood consistent with respect to $\mathscr{D}$. We begin with the following key lemma.

Lemma 39. *For any $n > 0$, let f_n denote the k_n-nearest neighbor classifier learned from $S \sim \mathscr{D}^n$. There exists some constant $\Delta > 0$ such that for all sufficiently large n, with probability at least $\frac{1}{2}$ over $S \sim \mathscr{D}^n$, there exists $x \in [0,1]$ with $\frac{1}{2} - \Delta \leq x \leq \frac{1}{2} - \frac{3\Delta}{4}$ and $f_n(x) = +1$.*

Proof. Let C be a constant such that $k_n \leq C \log n$ for all $2 \leq n < \infty$. Set Δ as

$$\frac{1}{2} \log_2 \frac{1}{1-2\Delta} + \frac{1}{2} \log_2 \frac{1}{1+2\Delta} < \frac{1}{C}. \tag{A.1}$$

Let $A \subset [0,1]$ denote the interval $[\frac{1}{2} - \Delta, \frac{1}{2} - \frac{3\Delta}{4}]$. For $S \sim \mathscr{D}^n$, with high probability, there exist at least $\frac{\Delta n}{8}$ instances x_i that are in A. Let us relabel these x_i as $x_1, x_2, \ldots, x_m$ as

$$\frac{1}{2} - \Delta \leq x_1 < x_2 < \cdots < x_m \leq \frac{1}{2} - \frac{3\Delta}{4}.$$

Next, suppose that for some i, at least half of $y_i, y_{i+1}, \ldots, y_{i+k_n-1}$ are $+1$. Then it follows that $f_n(x) = +1$ for $x = \frac{x_{i+k_n}+x_i}{2}$ because the k_n nearest neighbors of x are precisely $x_i, x_{i+1}, \ldots x_{i+k_n-1}$ (as a technical note we make x just slightly smaller to break the tie between x_i and x_{i+k_n}). To lower bound the probability that this occurs for some i, we partition $y_1, y_2, \ldots y_m$ into at least $\frac{m}{2k_n}$ disjoint groups each containing k_n consecutive values of y_i. We then bound the probability that each group will have at least $k_n/2$ +1s.

Consider any group of k_n y_is. We have that $\Pr[y_i] = +1 = \eta(x_i) = x_i \geq \frac{1}{2} - \Delta$. Since the variables y_i are independent (even conditioning on x_i), it follows that the probability that at least half of them are $+1$ is at least $\Pr[\text{Bin}(k_n, \frac{1}{2} - \Delta) \geq \frac{k_n}{2}]$. For simplicity, assume that k_n is even. Then using a standard lower bound for the tail of a binomial distribution (see, for example, Lemma 4.7.2 of [46]), we have that

$$\Pr[\text{Bin}(k_n, \frac{1}{2} - \Delta) \geq \frac{k_n}{2}] \geq \frac{1}{\sqrt{2k_n}} \exp(-k_n D(\frac{1}{2}||(\frac{1}{2} - \Delta)),$$

where $D(\frac{1}{2}||(\frac{1}{2} - \Delta)) = \frac{1}{2}\log_2 \frac{1}{1-2\Delta} + \frac{1}{2}\log_2 \frac{1}{1+2\Delta}$.

To simplify notation, let $D_\Delta = D(\frac{1}{2}||(\frac{1}{2} - \Delta))$. Then because we have $\frac{m}{2k_n}$ independent groups of y_is, we have that

$$\begin{aligned}
\Pr_{S\sim\mathscr{D}^n}[\exists x \in [\frac{1}{2} - \Delta, \frac{1}{2} - \frac{3\Delta}{4}] \text{ s.t. } f_n(x) = +1] &\geq 1 - (1 - \frac{1}{\sqrt{2k_n}} \exp(-k_n D_\Delta))^{\frac{m}{2k_n}} \\
&\geq 1 - \exp(-\frac{m}{2k_n\sqrt{2k_n}} e^{-k_n D_\Delta}) \\
&\geq 1 - \exp(-\frac{n\Delta}{(16C\log n)^{3/2}} e^{-CD_\Delta \log n}),
\end{aligned}$$

with the inequalities holding because $m \geq \frac{n\Delta}{8}$ and $k_n \leq C\log n$. By equation A.1, $CD_\Delta < 1$. Therefore, $\lim_{n\to\infty} \frac{n}{(2C\log n)^{3/2}} e^{-CD_\Delta \log n} = \infty$, which implies that for n sufficiently large,

$$\Pr_{S\sim\mathscr{D}^n}[\exists x \in [\frac{1}{2} - \Delta, \frac{1}{2} - \frac{3\Delta}{4}] \text{ s.t. } f_n(x) = +1] \geq \frac{1}{2},$$

as desired. □

We now complete the proof of Theorem 28.

Proof. (Theorem 28) Let Δ be as described in Lemma 39, and let $\kappa = \frac{1}{2}$. For all $x < \frac{1}{2}$, we have that $[x, \frac{2x}{3} + \frac{1}{6}] \subseteq V_x^\kappa$. This is because we can easily verify that all points inside that interval are closer to x than they are to $\frac{1}{2}$ (and consequently all points in $\mu^+ \cup \mu^{1/2}$) by factor of 2. It follows that for all $x \in [\frac{1}{2} - \frac{7\Delta}{8}, \frac{1}{2} - \Delta]$,

$$[\frac{1}{2} - \Delta, \frac{1}{2} - \frac{3\Delta}{4}] \subseteq V_x^\kappa.$$

However, applying Lemma 39, we know that with probability at least $\frac{1}{2}$, there exists some point $x' \in [\frac{1}{2} - \Delta, \frac{1}{2} - \frac{3\Delta}{4}]$ such that $f_n(x') = +1$. It follows that with probability at least $\frac{1}{2}$, f_n lacks astuteness at *all* $x \in [\frac{1}{2} - \frac{7\Delta}{8}, \frac{1}{2} - \Delta]$. Since this set of points has total probability mass $\Delta/8$, it follows that with probability at least $\frac{1}{2}$, there is a fixed gap between $A_{\mathscr{V}^\kappa}(f_n, \mathscr{D})$ and $A(g, \mathscr{D})$ (as they differ in a region of probability mass at least $\Delta/8$). This implies that k_n-nearest neighbors is not neighborhood consistent. □

A.2.3 Proof of Theorem 31

Let $\mathscr{D} = (\mu, \eta)$ is a distribution over $\mathbb{R}^d \times \{\pm 1\}$. We will use the following notation: let $\mathscr{D}^+ = \{x : \eta(x) > \frac{1}{2}\}$, $\mathscr{D}^- = \{x : \eta(x) < \frac{1}{2}$ and $\mathscr{D}_{1/2} = \{x : \eta(x) = \frac{1}{2}\}$. In particular, we have that $\mathscr{D}^+ = \mu^+, \mathscr{D}^- = \mu^-$ and $\mathscr{D}_{1/2} = \mu^{1/2}$. This notation serve will be convenient throughout this section since it allows us to avoid overloading the symbol μ.

To show that an algorithm is neighborhood consistent with respect to $\mathscr{D}$, we must show that for any $0 < \kappa < 1$, the astuteness with respect to $\mathscr{V}^\kappa$ converges towards the accuracy of the Bayes optimal. To this end, we fix any $0 < \kappa < 1$ and consider $\mathscr{V}^\kappa$.

For our proofs, it will be useful to have the additional assumption that the robustness regions, V_x^κ are *closed*. To obtain this, we let $\mathscr{U} = \{U_x\}$ where $U_x = \overline{V_x^\kappa}$. Each U_x is the closure of the corresponding V_x^κ, and in particular we have $V_x^\kappa \subset U_x$. Because of this, it will suffice for us to consider $A_{\mathscr{U}}$ as opposed to $A_{\mathscr{V}^\kappa}$ since $A_{\mathscr{U}}(f, \mathscr{D}) \leq A_{\mathscr{V}^\kappa}(f, \mathscr{D})$ for all classifiers f.

We now begin by first proving several useful properties of $\mathscr{U}$ that we will use throughout this entire section.

Lemma 40. *The collection of sets $\mathscr{U} = \{U_x\}$ defined as $U_x = \overline{V_x^\kappa}$ satisfies the following properties.*

1. *U_x is closed for all x.*
2. *if $x \in \mathscr{D}^+$, for all $x' \in U_x$, $\rho(x,x') < \rho(\mathscr{D}^+ \cup \mathscr{D}_{1/2}, x')$.*
3. *if $x \in \mathscr{D}^-$, for all $x' \in U_x$, $\rho(x,x') < \rho(\mathscr{D}^- \cup \mathscr{D}_{1/2}, x')$.*
4. *$U_x = \{x\}$ for all $x \in \mathscr{D}_{1/2}$.*
5. *U_x is bounded for all x.*

Here $\mu^+, \mu^-, \mu^{1/2}$ are as described in Definition 22.

Proof. Property (1) is given the by definition, and properties (2), (3) follow from the fact that κ is strictly less than 1. In particular, the distance function ρ is continuous and consequently all limit points of a set have distances that are limits of distances within the set. Property (4) is since $V_x^\kappa = \{x\}$ for all $x \in \mathscr{D}_{1/2}$.

Finally, property (5) follows from the fact that $\kappa < 1$. As x gets arbitrarily far away from x the ratio of its distance to x with its distance to μ^- gets arbitrarily close to 1, and consequently there is some maximum radius R so that $V_x^\kappa \subset B(x,R)$. Since $B(x,R)$ is closed, it follows that $U_x \subset B(x,R)$ as well. □

Next, fix W as a weight function and t_n is a sequence of positive integers such that the conditions of Theorem 31 hold, that is:

1. W is consistent (with resp. to accuracy) with resp. to $\mathscr{D}$.
2. For any $0 < p < 1$, $\lim_{n\to\infty} E_{S\sim\mathscr{D}^n}[\sup_{x\in\mathbb{R}^d} \sum_1^n w_i^S(x) 1_{\rho(x,x_i) > r_p(x)}] = 0$.

3. $\lim_{n\to\infty} E_{S\sim D^n}[t_n \sup_{x\in\mathbb{R}^d} w_i^S(x)] = 0.$

4. $\lim_{n\to\infty} E_{S\sim D^n} \frac{\log T(W,S)}{t_n} = 0.$

Finally, we will also make the additional assumption that $\mathscr{D}$ has infinite support. Cases where $\mathscr{D}$ has finite support can be somewhat trivially handled: when the sample size goes to infinity, we will have perfect labels for every point in the support, and consequently condition 2. will ensure that any $x' \in V_x^\kappa$ is labeled according to the label of x.

We also use the following notation. For any classifier $f : \mathbb{R}^d \to \{\pm 1\}$, we let

$$\mathscr{D}_f^+ = \{x : f(x' = +1 \text{ for all } x' \in U_x\}, \text{ and } \mathscr{D}_f^- = \{x : f(x' = -1 \text{ for all } x' \in U_x\}. \tag{A.2}$$

These sets represent the examples that f robustly labels as $+1$ and -1 respectively. These sets are useful since they allows us to characterize the astuteness of f, which we do with the following lemma.

Lemma 41. *For any classifier $f : \mathbb{R}^d \to \{\pm 1\}$, we have*

$$A_{\mathscr{U}}(f, \mathscr{D}) \geq A(g, \mathscr{D}) - \mu(\mathscr{D}^+ \setminus \mathscr{D}_f^+) - \mu(D^- \setminus \mathscr{D}_f^-),$$

where g denotes the Bayes optimal classifier.

Proof. By property 4 of Lemma 40, $U_x = \{x\}$ for all $x \in \mathscr{D}_{1/2}$. Consequently, if $x \in \mathscr{D}_{1/2}$, there is a $\frac{1}{2}$ chance that any classifier is astute at (x, y). Using this along with the definition of astuteness, we see that

$$\begin{aligned} A_{\mathscr{U}}(f, \mathscr{D}) &= \Pr_{(x,y)\sim\mathscr{D}}[f(x') = y \text{ for all } x' \in U_x] \\ &= \Pr_{(x,y)\sim\mathscr{D}}[y = +1 \text{ and } x \in (D^+ \cap D_f^+)] + \Pr_{(x,y)\sim\mathscr{D}}[y = -1 \text{ and } x \in (D^- \cap D_f^-)] + \frac{1}{2}\Pr_{(x,y)\sim\mathscr{D}}[x \in \mathscr{D}_{1/2}] \end{aligned}$$

However, observe by the definitions of $\mathscr{D}^+, \mathscr{D}^-$ and $\mathscr{D}_{1/2}$ that

$$A(g,\mathscr{D}) = \Pr_{(x,y)\sim\mathscr{D}}[y=+1 \text{ and } x\in D^+] + \Pr_{(x,y)\sim\mathscr{D}}[y=-1 \text{ and } x\in D^-] + \frac{1}{2}\Pr_{(x,y)\sim\mathscr{D}}[x\in\mathscr{D}_{1/2}].$$

Substituting this, we find that

$$\begin{aligned} A_{\mathscr{U}}(f,\mathscr{D}) &\geq A(g,\mathscr{D}) - \Pr_{(x,y)\sim\mathscr{D}}[x\in(D^+\setminus D_f^+)] - \Pr_{(x,y)\sim\mathscr{D}}[x\in(D^-\setminus D_f^-)] \\ &= A(g,\mathscr{D}) - \mu(\mathscr{D}^+\setminus\mathscr{D}_f^+) - \mu(D^-\setminus\mathscr{D}_f^-), \end{aligned}$$

as desired. □

Lemma 41 shows that to understand how W_S converges in astuteness, it suffices to understand how the regions $\mathscr{D}^+_{W_S}$ and $\mathscr{D}^-_{W_S}$ converge towards D^+ and D^- respectively. This will be our main approach for proving Theorem 31. Due to the inherent symmetry between $+$ and $-$, we will focus on showing how the region $\mathscr{D}^+_{W_S}$ converges towards D^+. The case for $-$ will be analogous. To that end, we have the following key definition.

Definition 42. *Let $p,\Delta>0$. We say $x\in\mathscr{D}^+$ is (p,Δ)-**covered** if for all $x'\in U_x$ and for all $x''\in B(x',r_p(x'))\cap supp(\mu)$, $\eta(x'')>\frac{1}{2}+\Delta$. Here r_p denotes the probability radius (Definition 29). We also let $\mathscr{D}^+_{p,\Delta}$ denote the set of all $x\in\mathscr{D}^+$ that are (p,Δ)-covered.*

If x is (p,Δ)-covered, it means that for all $x'\in U_x$, there is a set of points with measure p around x' that are both close to x', and likely (with at least probability $\frac{1}{2}+\Delta$) to be labeled as $+1$. Our main idea will be to show that if x is (p,Δ) covered and n is sufficiently large, x is likely to be in $\mathscr{D}^+_{W_S}$.

We begin this process by first showing that all x are (p,Δ)-covered for some p,Δ. To do so, it will be useful to have one more piece of notation which we will also use throughout the rest of the section. We let

$$\mathscr{D}^-_{1/2} = \mathscr{D}^-\cup\mathscr{D}_{1/2} = supp(\mu)\setminus\mathscr{D}^+.$$

This set will be useful, since Lemma 40 implies that for all $x \in \mathscr{D}^+$ and for all $x' \in U_x$, $\rho(x,x') < \rho(\mathscr{D}^-_{1/2}, x')$. We now return to showing that all x are $(p,\Delta$-covered for some p,Δ.

Lemma 43. *For any $x \in \mathscr{D}^+$, there exists $p,\Delta > 0$ such that x is (p,Δ)-covered.*

Proof. Fix any x. Let $f : U_x \to \mathbb{R}$ be the function defined as $f(x') = \rho(x', \mathscr{D}^-_{1/2}) - \rho(x',x)$. Observe that f is continuous. By assumption, U_x is closed and bounded, and consequently must attain its minimum. However, by Lemma 40, we have that $f(x') > 0$ for all $x' \in U_x$. it follows that $\min_{x' \in U_x} f(x') = \gamma$ where $\gamma > 0$.

Next, let $p = \mu(B(x, \gamma/2))$. $p > 0$ since $x \in supp(\mu)$. Observe that for any $x' \in U_x$, $r_p(x') \leq \rho(x,x') + \gamma/2$, where, $r_p(x')$ denotes the probability radius of x'. This is because $B(x', (\rho(x,x') + \gamma/2))$ contains $B(x, \gamma/2)$ which has probability mass p. It follows that for any $x' \in U_x$, $\rho(x', \mathscr{D}^-_{1/2}) \geq r_p(x') + \gamma/2$. Motivated by this observation, let A be the region defined as

$$A = \bigcup_{x' \in U_x} B(x', r_p(x')).$$

Then by our earlier observation, we have that $\rho(A, \mathscr{D}^-_{1/2}) \geq \frac{\gamma}{2}$. Since distance is continuous, it follows that $\rho(\overline{A}, \mathscr{D}^-_{1/2}) \geq \frac{\gamma}{2}$ as well, where $\overline{A}$ denotes the closure of A.

This means that for any $x'' \in \overline{A} \cap supp(\mu)$, $\eta(x'') > \frac{1}{2}$, since otherwise $\rho(\overline{A}, \mathscr{D}^-_{1/2})$ would equal 0 (as the two sets would literally intersect). Finally, $supp(\mu)$ is a closed set (see Appendix A.3.1), and thus $\overline{A} \cap supp(\mu)$ is closed as well. Since η is continuous (by assumption from Definition 27), it follows that η must maintain its minimum value over $\overline{A} \cap supp(\mu)$. It follows that there exists $2\Delta > 0$ such that $\eta(x'') \geq \frac{1}{2} + 2\Delta > \frac{1}{2} + \Delta$ for all $x'' \in \overline{A} \cap supp(\mu)$.

Finally, by the definition of A, for all $x' \in U_x$, $B(x', r_p(x')) \subset A$. It consequently follows from the definition that x is (p,Δ)-covered, as desired. □

While the previous lemma show that some p,Δ cover any $x \in \mathscr{D}^+$, this does not necessarily mean that there are some fixed p,Δ that cover *all* $x \in \mathscr{D}^+$. Nevertheless, we can show that

this is almost true, meaning that there are some p, Δ that cover *most* $x \in \mathscr{D}^+$. Formally, we have the following lemma.

Lemma 44. *For any $\varepsilon > 0$, there exists p, Δ such that $\mu(\mathscr{D}^+ \setminus \mathscr{D}^+_{p,\Delta}) < \varepsilon$, where $\mathscr{D}^+_{p,\Delta}$ is as defined in Definition 42.*

Proof. Observe that if x is (p,Δ)-covered, then it is also (p',Δ')-covered for any $p' < p$ and $\Delta' < \Delta$. This is because $B(x', r_{p'}(x')) \subset B(x', r_p(x))$ and $\frac{1}{2} + \Delta > \frac{1}{2} + \Delta'$. Keeping this in mind, define

$$\mathscr{A} = \{\mathscr{D}^+_{1/i,1/j} : i, j \in \mathbb{N}\}.$$

For any $x \in \mathscr{D}^+$, by Lemma 43 and our earlier observation, there exists $A \in \mathscr{A}$ such that $x \in A$. It follows that $\cup_{A \in \mathscr{A}} A = \mathscr{D}^+$. By applying Lemma 61, we see that there exists a finite subset of $\mathscr{A}$, $\{A_1, \ldots, A_m\}$ such that

$$\mu(A_1 \cup \cdots \cup A_m\}) > \mu(\mathscr{D}^+) - \varepsilon.$$

Let $A_k = \mathscr{D}^+_{1/i_k,1/j_k}$ for $1 \leq k \leq m$. From our previous observation once again, we see that $\cup A_i \subset \mathscr{D}^+_{1/I,1/J}$ where $I = \max(i_k)$ and $J = \max(j_k)$. It follows that setting $p = 1/I$ and $\Delta = 1/J$ suffices. □

Recall that our overall goal is to show that if x is (p,Δ)-covered, n is sufficiently large, then x is very likely to be in $\mathscr{D}^+_{W_S}$ (defined in equation A.2). To do this, we will need to find sufficient conditions on S for x to be in W_S. This requires the following definitions, that are related to *splitting numbers* (Definition 30).

Definition 45. *Let $x \in \mathbb{R}^d$ be a point, and let $S = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ be a training set sampled from $\mathscr{D}^n$. For $0 \leq \alpha$, $0 \leq \beta \leq 1$, and $0 < \Delta < \frac{1}{2}$, we define*

$$W^{\Delta,S}_{x,\alpha,\beta} = \{i : \rho(x, x_i) \leq \alpha, w^S_i(x) \geq \beta, \eta(x_i) > \frac{1}{2} + \Delta\}.$$

Definition 46. *Let $0 < \Delta < \frac{1}{2}$, and let $S = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ be a training set sampled from $\mathscr{D}^n$. Then we let*

$$W^{\Delta,S} = \{W^{\Delta,S}_{x,\alpha,\beta} : x \in \mathbb{R}^d, 0 \leq \alpha, 0 \leq \beta \leq 1\}.$$

These convoluted looking sets will be useful for determining the behavior of W_s at some $x \in \mathscr{D}^+_{p,\Delta}$. Broadly speaking, the idea is that if every set of indices $R \subset W^{\Delta,S}$ is relatively well behaved (i.e. the number of y_is that are $+1$ is close to $(|R|(\frac{1}{2}+\Delta)$, the expected amount), then $W_s(x') = +1$ for all $x' \in U_x$. Before showing this, we will need a few more lemmas.

Lemma 47. *Fix any $\delta > 0$ and let $0 < \Delta < \frac{1}{2}$. There exists N such that for all $n > N$ the following holds. With probability $1 - \delta$ over $S \sim \mathscr{D}^n$, for all $R \in W^{\Delta,S}$ with $|R| > t_n$, $\frac{1}{|R|}\sum_{i \in R} y_i \geq \Delta$*

Proof. The key idea is to observe that the set $W^{\Delta,S}$ and the value $T(W,S)$ are completely determined by $\{x_1, \ldots, x_n\}$. This is because weight functions choose their weights only through dependence on $x_1, \ldots, x_n$. Consequently, we can take the equivalent formulation of first drawing $x_1, \ldots, x_n \sim \mu^n$, and then drawing y_i independently according to $y_i = 1$ with probability $\eta(x_1)$ and 0 with probability $1 - \eta(x_i)$. In particular, we can treat $y_1, \ldots, y_n$ as independent from $W^{\Delta,S}$ and $T(W,S)$ conditioning on $x_1, \ldots, x_n$.

Fix any $x_1, \ldots, x_n$. First, we see that $|W^{\Delta,S}| \leq T(W,S)$. This is because $W^{\Delta,S}_{x,\alpha,\beta}$ is a subset that is uniquely defined by $W_{x,\alpha,\beta}$ (see Definitions 45 and 30). Second, for any $R \in W^{\Delta,S}$, observe that for all $i \in R$, y_i is a binary variable in $[-1,1]$ with expected value at least $(\frac{1}{2}+\Delta) - (\frac{1}{2}-\Delta) = 2\Delta$ (again by the definition). It follows that if $|R| \geq t_n$, by Hoeffding's inequality

$$\Pr_{y_1 \ldots y_n}[\sum_{i \in R} y_i < \Delta] \leq \exp\left(-\frac{2|R|^2\Delta^2}{4|R|}\right) \leq \exp\left(-\frac{t_n\Delta^2}{2}\right).$$

Since there at most $T(W,S)$ sets R, it follows that

$$\Pr_{y_1 \ldots y_n}[\sum_{i \in R} y_i < \Delta \text{ for some } R \in W^{\Delta,S} \text{ with } |R| > t_n] \leq T(W,S)\exp\left(-\frac{t_n\Delta^2}{2}\right).$$

However, by condition 4. of Theorem 31, it is not difficult to see that this quantity has expectation that tends to 0 as $n \to \infty$ (unless $T(W,S)$ uniformly equals 1, but this degenerate case can easily be handled on its own). Thus, for any $\delta > 0$, it follows that there exists N such that for all $n > N$, with probability at least $1 - \frac{\delta}{2}$, $T(W,S)\exp\left(-\frac{t_n\Delta^2}{2}\right) \leq \frac{\delta}{2}$. This value of N consequently suffices for our lemma. □

We now relate $\mathscr{D}^+_{W_S}$ (Equation A.2) to $W^{\Delta,S}$ as well as the conditions of Theorem 31.

Lemma 48. *Let $S = \{(x_1,y_1),\ldots,(x_n,y_n)\}$ and let $0 < \Delta \leq \frac{1}{2}$ and $0 < p < 1$ such that the following conditions hold.*

1. *For all $R \in W^{\Delta,S}$ with $|R| > t_n$, $\frac{1}{|R|}\sum_{i\in R} y_i \geq \Delta$.*
2. *$\sup_{x\in\mathbb{R}^d} \sum_1^n w_i^S(x) 1_{\rho(x,x_i) > r_p(x)} < \frac{\Delta}{5}$.*
3. *$t_n \sup_{x\in\mathbb{R}^d} w_i^S(x) < \frac{\Delta}{5}$.*

Then $\mathscr{D}^+_{p,\Delta} \subseteq \mathscr{D}^+_{W_S}$.

Proof. Let $x \in \mathscr{D}^+_{p,\Delta}$, and let $x' \in U_x$ be arbitrary. It suffices to show that $W_S(x') = +1$ (as x, x' were arbitrarily chosen). From the definition of W_S, this is equivalent to showing that $\sum_1^n w_i^S(x')y_i > 0$. Thus, our strategy will be to lower bound this sum using the conditions given in the lemma statement.

We first begin by simplifying notation. Since S and x' are both fixed, we use w_i to denote $w_i^S(x')$. Since n is fixed, we will also use t to denote t_n. Next, suppose that $|\{x_1,\ldots,x_n\} \cap B(x', r_p(x'))| = k$. Without loss of generality, we can rename indices such that $\{x_1,\ldots,x_n\} \cap B(x', r_p(x')) \cap B(x', r_p(x')) = \{x_1,\ldots,x_k\}$, and $w_1 \geq w_2 \geq \cdots \geq w_k$.

Let $Y_j = \sum_{i=1}^j y_i$. Our main idea will be to express the sum in terms of these Y_js as

follows.

$$\begin{aligned}\sum_{1}^{n} w_i y_i &= \sum_{1}^{k} w_i y_i + \sum_{k+1}^{n} w_i y_i \\ &= w_k Y_k + (w_{k-1} - w_k) Y_{k-1} + \cdots + (w_{t+1} - w_{t+2}) Y_{t+1} + \sum_{i=1}^{t} (w_i - w_{t+1}) y_i + \sum_{k+1}^{n} w_i y_i \\ &= \underbrace{w_k Y_k + \sum_{i=t+1}^{k-1} (w_i - w_{i+1}) Y_i}_{\alpha} + \underbrace{\sum_{i=1}^{t} (w_i - w_{t+1}) y_i}_{\beta} + \underbrace{\sum_{k+1}^{n} w_i y_i}_{\tau}.\end{aligned}$$

We now bound α, β and τ in terms of Δ by using the conditions given in the lemma. We begin with β and τ, which are considerably easier to handle.

For β, we have that

$$\beta = \sum_{i=1}^{t} (w_i - w_{t+1}) y_i \geq \sum_{i=1}^{t} (w_i - w_{t+1})(-1) \geq -t w_1.$$

By condition 2 of the lemma, we see that $t w_1 < \frac{\Delta}{5}$, which implies that $\beta \geq -\frac{\Delta}{5}$.

For γ, we have that $\gamma = \sum_{k+1}^{n} w_i y_i \geq -\sum_{k+1}^{n} w_i$. However, for all $k+1 \leq i \leq n$, by definition of k, $\rho(x', x_i) > r_p(x')$. It follows from condition 3 of the lemma that $\gamma \geq -\frac{\Delta}{5}$.

Finally, we handle α. Recall that x is (p, Δ)-covered. It follows that for all $x'' \in supp(\mu) \cap B(x', r_p(x'))$, $\eta(x'') > \frac{1}{2} + \Delta$. Thus, by the definition of k, $\eta(x_i) > \frac{1}{2} + \Delta$ for $1 \leq i \leq k$. It follows that if $w_i > w_{i+1}$ or $i = k$, then

$$\begin{aligned}W^{\Delta,S}_{x', r_p(x'), w_i} &= \{j : \rho(x', x_j) \leq r_p(x'), w_j \geq w_i, \eta(x_j) > \frac{1}{2} + \Delta\} \\ &= \{1, \ldots, i\}.\end{aligned}$$

This implies that $\{1, \ldots, i\} \in W^{\Delta,S}$, and consequently that $Y_i \geq i\Delta$, from condition 1 of the lemma. It follows that for all $t < i \leq k$, $(w_i - w_{i+1}) Y_i \geq i(w_i - w_{i+1})\Delta$, and that $w_k Y_k \geq k w_k \Delta$.

Substituting these, we find that

$$\begin{aligned}
\alpha &= w_k Y_k + \sum_{i=t+1}^{k-1} (w_i - w_{i+1}) Y_i \\
&\geq k w_k \Delta + \sum_{i=t+1}^{k-1} i(w_i - w_{i+1})\Delta \\
&= w_k\Delta + w_{k-1}\Delta + \cdots + w_{t+1}\Delta + (t+1)w_{t+1}\Delta. \\
&\geq (1 - \sum_{1}^{t} w_i - \sum_{k+1}^{n} w_i)\Delta \\
&\geq (1 - \frac{2\Delta}{5})\Delta \\
&\geq (\frac{4\Delta}{5}),
\end{aligned}$$

with the last inequalities holding from the arguments given for β and γ along with the fact that $0 < \Delta \leq \frac{1}{2}$. Finally, substituting these, we find that $\alpha + \beta + \gamma \geq \frac{4\Delta}{5} - \frac{2\Delta}{5} = \frac{2\Delta}{5} > 0$, as desired. □

We are now ready to prove the key lemma that forms one half of the main theorem (the other half corresponding to $\mathscr{D}^{-}_{W_S}$).

Lemma 49. *Let $\delta, \varepsilon > 0$. There exists N such that for all $n > N$, with probability $1 - \delta$ over $S \sim \mathscr{D}^n$, $\mu(\mathscr{D}^{+} \setminus \mathscr{D}^{+}_{W_S}) < \varepsilon$.*

Proof. First, by Lemma 44, let $0 < p$ and $0 < \Delta$ be such that $\mu(\mathscr{D}^{+} \setminus \mathscr{D}^{+}_{p,\Delta}) < \varepsilon$. By combining Lemma 47, condition 3 of Theorem 31, and condition 2 of Theorem 31 respectively, we see that there exists N such that for all $n > N$, the following hold:

1. With probability at least $1 - \frac{\delta}{3}$ over $S \sim \mathscr{D}^n$, for all $R \in W^{\Delta,S}$ with $|R| > t_n$, $\frac{1}{|R|}\sum_{i \in R} y_i \geq \Delta$.
2. With probability at least $1 - \frac{\delta}{3}$ over $S \sim \mathscr{D}^n$, $\sup_{x \in \mathbb{R}^d} \sum_1^n w_i^S(x) 1_{\rho(x,x_i) > r_p(x)} < \frac{\Delta}{5}$.
3. With probability at least $1 - \frac{\delta}{3}$ over $S \sim \mathscr{D}^n$, $t_n \sup_{x \in \mathbb{R}^d} w_i^S(x) < \frac{\Delta}{5}$.

By a union bound, this implies that p, Δ, S satisfy the conditions of Lemma 48 with probability at least $1 - \delta$. Thus, applying the Lemma, we see that with probability $1 - \delta$, $\mathscr{D}^{+}_{p,\Delta} \subset \mathscr{D}^{+}_{W_S}$. This

immediately implies our claim. □

By replicating all of the work in this section for $\mathscr{D}^-$ and $\mathscr{D}^-_{p,\Delta}$, we can similarly show the following:

Lemma 50. *Let $\delta, \varepsilon > 0$. There exists N such that for all $n > N$, with probability $1 - \delta$ over $S \sim \mathscr{D}^n$, $\mu(\mathscr{D}^- \setminus \mathscr{D}^-_{W_S}) < \varepsilon$.*

Combining these two lemmas with Lemma 41 immediately implies that for all $\delta, \varepsilon > 0$, there exists N such that for all $n > N$, with probability $1 - \delta$ over $S \sim \mathscr{D}^n$,

$$A_{\mathscr{U}}(W_S, \mathscr{D}) \geq A(g, \mathscr{D}) - \varepsilon.$$

Since $V_x^\kappa \subset U_x$ and since κ was arbitrary, this implies Theorem 31, which completes our proof.

A.2.4 Proof of Corollary 32

Recall that k_n-nearest neighbors can be interpreted as a weight function, in which $w_i^S(x) = \frac{1}{k_n}$ if x_i is one of the k_n closest points to x, and 0 otherwise. Therefore, it suffices to show that the conditions of Theorem 31 are met.

We let W denote the weight function associated with k_n-nearest neighbors.

Lemma 51. *W is consistent.*

Proof. It is well known (for example [35]) that k_n-nearest neighbors is consistent for $\lim_{n\to\infty} k_n = \infty$ and $\lim_{n\to\infty} \frac{k_n}{n} = 0$. These can easily be verified for our case. □

Lemma 52. *For any $0 < p < 1$, $\lim_{n\to\infty} \mathbb{E}_{S\sim\mathscr{D}^n}[\sup_{x\in\mathbb{R}^d} \sum_1^n w_i^S(x) 1_{\rho(x,x_i) > r_p(x)}] = 0$.*

Proof. It suffices to show that for n sufficiently large, all k_n-nearest neighbors of x are located inside $B(x, r_p(x))$ for all $x \in \mathbb{R}^d$. We do this by using a VC-dimension type argument to show that all balls $B(x,r)$ contain a number of points from $S \sim \mathscr{D}^n$ that is close to their expectation.

For $x \in \mathbb{R}^d$ and $r \geq 0$, let $f_{x,r}$ denote the $0-1$ function defined as $f_{x,r}(x') = 1_{x' \in B(x,r)}$. Let $F = \{f_{x,r} : x \in \mathbb{R}^d, r \geq 0\}$ denote the class of all such functions. It is well known that the VC dimension of F is at most $d+2$.

For $f \in F$, let $\mathbb{E}f$ denote $\mathbb{E}_{(x',y)\sim\mathscr{D}} f(x')$ and $\mathbb{E}_n f$ denote $\frac{1}{n}\sum_1^n f(x_i)$, where $\mathbb{E}_n f$ is defined with respect to some sample $S \sim \mathscr{D}^n$. By the standard generalization result of Vapnik and Chervonenkis (see [47] for a proof), we have that with probability $1-\delta$ over $S \sim \mathscr{D}^n$,

$$-\beta_n \sqrt{\mathbb{E}f} \leq \mathbb{E}f - \mathbb{E}_n f \leq \beta_n \sqrt{\mathbb{E}f} \tag{A.3}$$

holds for all $f \in F$, where $\beta_n = \sqrt{(4/n)((d+2)\ln 2n + \ln(8/\delta)}$.

Suppose n is sufficiently large so that $\beta_n \leq \frac{p}{2}$ and $\frac{k_n}{n} < \frac{p}{2}$, and suppose that equation A.3 holds. Pick any $x \in \mathbb{R}^d$ and consider $f_{x,r}$ where $r > r_p(x)$. This implies $\mathbb{E}f_{x,r} \geq p$. Then by equation A.3, we see that $\mathbb{E}_n f \geq \frac{p}{2}$. This implies that all k_n nearest neighbors of x are in the ball $B(x,r)$, and that consequently $\sum_1^n w_i^S(x) 1_{\rho(x,x_i)>r} = 0$. Because this holds for all x, r with $x \in \mathbb{R}^d$ and $r > r_p(x)$, it follows that equation 2 implies that

$$\sup_{x \in X} \sum_1^n w_i^S(x) 1_{\rho(x,x_i) > r_p(x)} = 0.$$

Because equation A.3 holds with probability at least $1-\delta$, and δ can be made arbitrarily small, the desired claim follows. □

Let $t_n = \sqrt{dk_n \log n}$.

Lemma 53. $\lim_{n\to\infty} E_{S\sim D^n}[t_n \sup_{x\in\mathbb{R}^d} w_i^S(x)] = 0$.

Proof. Let $S \sim \mathscr{D}^n$. By the definition of k_n nearest neighbors, $\sup_{x\in\mathbb{R}^d} w_i^S(x) = \frac{1}{k_n}$. Therefore, $t_n \sup_{x\in\mathbb{R}^d} w_i^S(x) = \sqrt{\frac{d\log n}{k_n}}$. By assumption 2. of corollary 32, $\lim_{n\to\infty} \frac{d\log n}{k_n} = 0$, which implies that

$$\lim_{n\to\infty} \mathbb{E}_{S\sim D^n}[t_n \sup_{x\in\mathbb{R}^d} w_i^S(x)] = \lim_{n\to\infty} \sqrt{\frac{d\log n}{k_n}} = \lim_{n\to\infty} \frac{d\log n}{k_n} = 0,$$

as desired. □

Lemma 54. $\lim_{n\to\infty} E_{S\sim D^n} \frac{\log T(W,S)}{t_n} = 0.$

Proof. For $S \sim \mathscr{D}^n$, recall that $T(W,S)$ was defined as

$$T(W,S)|\{W_{x,\alpha,\beta} : x \in \mathbb{R}^d, 0 \leq \alpha, 0 \leq \beta \leq 1\}|,$$

where $W_{x,\alpha,\beta}$ denotes

$$W_{x,\alpha,\beta} = \{i : \rho(x,x_i) \leq \alpha, w_i^S(x) \geq \beta\}.$$

Our goal will to be upper bound $\log T(W,S)$.

To do so, we first need a tie-breaking mechanism for k_n-nearest neighbors. For each $x_i \in S$, we independently sample $z_i \in [0,1]$ from the uniform distribution. We then tie break based upon the value of z_i, i.e. if $\rho(x,x_i) = \rho(x,x_j)$, we say that x_i is closer to x than x_j if $z_i < z_j$. With probability 1, no two values z_i, z_j will be equal, so this ensures that this method always works.

Let $A_{x,\alpha} = \{i : \rho(x,x_i) \leq \alpha\}$ and let $B_{x,c} = \{i : z_i \leq c\}$. The key observation is that for any α, β, $W_{x,\alpha,\beta} = A_{x,\alpha} \cap B_{x,c}$ for some value of c. This can be seen by noting that the nearest neighbors of x are uniquely determined by $\rho(x,x_i)$ and z_i. Therefore, it suffices to bound $|A = A_{x,\alpha} : x \in \mathbb{R}^d, \alpha \geq 0\}|$ and $|B = \{B_{x,c} : x \in \mathbb{R}^d, c \geq 0\}|$.

To bound $|A|$, observe that the set of closed balls in $\mathbb{R}^d$ has VC-dimension at most $d+2$. Thus by Sauer's lemma, there are at most $O(n^{d+2}$ subsets of $\{x_1, x_2, \ldots, x_n\}$ that can be obtained from closed balls. Thus $|A| \leq O(n^{d+2}$.

To bound $|B|$, we simply note that $B_{x,c}$ consists of all i for which $z_i \leq c$. Since the z_i can be sorted, there are at most $n+1$ such sets. Thus $|B| \leq n+1$.

Combining this, we see that $T(W,S) \leq |A||B| \leq O(n^{d+3})$. Finally, we see that

$$\lim_{n\to\infty} \frac{\log T(W,S)}{t_n} = \lim_{n\to\infty} \frac{O(d\log n)}{\sqrt{k_n d \log n}} = \lim_{n\to\infty} \sqrt{\frac{O(d\log n)}{k_n}} = 0,$$

with the last inequality holding by condition 2. of Corollary 32.

□

Finally, we note that Corollary 32 is an immediate consequence of the previous 4 lemmas as we can simply apply Theorem 31.

A.2.5 Proof of Corollary 33

Let W be a kernel classifier constructed from K and h_n such that the conditions of Corollary 33 hold: that is,

1. $K : [0,\infty) \to [0,\infty)$ is decreasing and satisfies $\int_{\mathbb{R}^d} K(x)dx < \infty$.
2. $\lim_{n\to\infty} h_n = 0$ and $\lim_{n\to\infty} nh_n^d = \infty$.
3. For any $c > 1$, $\lim_{x\to\infty} \frac{K(cx)}{K(x)} = 0$.
4. For any $x \geq 0$, $\lim_{n\to\infty} \frac{n}{\log n} K(\frac{x}{h_n}) = \infty$.

It suffices to show that the conditions of Theorem 31 are met for W. Before doing this, we will describe one additional assumption we make for this case.

Additional Assumption:

We assume that $\mathscr{D}, \mathscr{U}$ are such that there exists some compact set $\mathscr{X} \subset \mathbb{R}^d$ such that for all $x \in supp(\mu)$, $U_x \subset \mathscr{X}$. This is primarily for convenience: observe that any distribution can be approximated arbitrarily closely by distributions satisfying these properties (as each U_x is bounded by assumption). Importantly, because of this, we will note that it is possible for conditions 2. and 3. of Theorem 31 to be relaxed to taking supremums over $\mathscr{X}$ rather than $\mathbb{R}^d$. This is because in our proof, we only ever used these conditions in their restriction to $\bigcup_{x\in supp(\mu)} \bigcup x' \in U_x B(x', r_p(x'))$.

Using this assumption, we return to proving the corollary.

Lemma 55. *W is consistent with respect to $\mathscr{D}$.*

Proof. Condition 1. of Corollary 33 imply that K is a regular kernel. This together with Condition 2. implies that W is consistent: a proof can be found in [9]. □

To verify the second condition, it will be useful to have the following definition.

Definition 56. *For any $p, \varepsilon > 0$ and $x \in \mathscr{X}$, define r_p^ε as*

$$r_p^\varepsilon(x) = \sup\{r : \mu(B(x,r)) - \mu(B(x, r_p(x)) \leq \varepsilon\}.$$

Lemma 57. *For any $p, \varepsilon > 0$, there exists a constant $c_p^\varepsilon > 1$ such that $\frac{r_p^\varepsilon(x)}{r_p(x)} \geq c_p^\varepsilon$ for all $x \in \mathscr{X}$, where we set $\frac{r_p^\varepsilon(x)}{r_p(x)} = \infty$ if $r_p(x) = 0$.*

Proof. The basic idea is to use the fact that $\mathscr{X}$ is compact. Our strategy will be to analyze the behavior of $\frac{r_p^\varepsilon(x)}{r_p(x)}$ over small balls $B(x_0, r)$ centered around some fixed x_0, and then use compactness to pick some finite set of balls $B(x_0, r)$. This must be done carefully because the function $x \to \frac{r_p^\varepsilon(x)}{r_p(x)}$ is not necessarily continuous.

Fix any $x_0 \in \mathscr{X}$. First, observe that $r_p^\varepsilon(x_0) > r_p(x_0)$. This is because $B(x_0, r_p(x_0)) = \cap_{r > r_p(x_0)} B(x_0, r)$, and consequently $\lim_{r \downarrow r_p(x_0)} \mu(B(x_0, r)) = \mu(B(x_0, r_p(x_)))$.

Next, define

$$s_p^\varepsilon(x) = \inf\{r : \mu(B(x, r_p(x)) - \mu(B(x,r)) \leq \varepsilon\}.$$

We can similarly show that $r_p(x_0) > s_p^\varepsilon(x_0)$.

Finally, define

$$r_0 = \frac{1}{3} \min(r_p^\varepsilon(x_0) - r_p(x_0), r_p(x_0) - s_p^\varepsilon(x_0)).$$

Consider any $x \in B^o(x_0, r_0)$ where B^o denotes the open ball, and let $\alpha = \rho(x_0, x)$. Then we have the following.

1. $r_p(x) \leq r_p(x_0) + \alpha$. This holds because $B(x, r_p(x_0) + \alpha)$ contains $B(x_0, r_p(x_0))$, which has probability mass at least p.

2. $r_p(x) \geq r_p(x_0) - \alpha$. This holds because if $r_p(x) < r_p(x_0) - \alpha$, then there would exists $r < r_p(x_0)$ such that $\mu(B(x_0, r)) \geq p$ which is a contradiction.

3. $B(x_0, s_p^\varepsilon(x_0)) \subset B(x, r_p(x))$. This is just a consequence of the definition of r_0 and the previous observation.

By the definitions of r_p^ε and s_p^ε, we see that $\mu(B(x_0, r_p^\varepsilon(x_0)) - \mu(B(x_0, s_p^\varepsilon(x_0)) \leq 2\varepsilon$. By the triangle inequality, $B(x, r_p^\varepsilon(x_0) - \alpha) \subset B(x_0, r_p^\varepsilon(x_0))$ and $B(x_0, s_p^\varepsilon(x_0)) \subset B(x, r_p(x))$. it follows that

$$\mu(B(x, r_p^\varepsilon(x_0) - \alpha)) - \mu(B(x, r_p(x))) \leq 2\varepsilon,$$

which implies that $r_p^{2\varepsilon}(x) \geq r_p^\varepsilon(x_0) - \alpha$. Therefore we have the for all $x \in B(x_0, r_0)$,

$$\frac{r_p^{2\varepsilon}(x)}{r_p(x)} \geq \frac{r_p^\varepsilon(x_0) - \alpha}{r_p(x_0) + \alpha} \geq \frac{2r_p^\varepsilon(x_0) + r_p(x_0)}{r_p^\varepsilon(x_0) + 2r_p(x_0)}.$$

Notice that the last expression is a constant that depends only on x_0, and moreover, since $r_p^\varepsilon(x_0) > r_p(x_0)$, this constant is strictly larger than 1. Let us denote this as $c(x_0)$. Then we see that $\frac{r_p^{2\varepsilon}(x)}{r_p(x)} \geq c(x_0)$ for all $x \in B^o(x_0, r_0)$.

Finally, observe that $\{B^o(x_0, r_0) : x_0 \in \mathscr{X}\}$ forms an open cover of $\mathscr{X}$ and therefore has a finite sub-cover C. Therefore, taking $c = \min_{B^o(x_0, r_0) \in C} c(x_0)$, we see that $\frac{r_p^{2\varepsilon}(x)}{r_p(x)} \geq c > 1$ for all $x \in \mathscr{X}$. Because ε was arbitrary, the claim holds. □

Lemma 58. *For any* $0 < p < 1$, $\lim_{n\to\infty} \mathbb{E}_{S\sim\mathscr{D}^n}[\sup_{x\in\mathscr{X}} \sum_1^n w_i^S(x) 1_{\rho(x,x_i) > r_p(x)}] = 0$.

Proof. Fix $p > 0$, and fix any $\varepsilon, \delta > 0$. Pick n sufficiently large so that the following hold.

1. Let c_p^ε be as defined from Lemma 57.

$$\sup_{x\in\mathscr{X}} \frac{K(c_p^\varepsilon r_p(x)/h_n)}{K(r_p(x)/h_n)} < \delta. \tag{A.4}$$

This is possible because of conditions 2. and 3. of Corollary 33, and because the function $x \to r_p(x)$ is continuous.

2. With probability at least $1-\delta$ over $S \sim \mathscr{D}^n$, for all $r > 0$, and $x \in \mathscr{X}$,

$$|\mu(B(x,r)) - \frac{1}{n}\sum_1^n 1_{x_i \in B(x,r)}| \leq \varepsilon. \tag{A.5}$$

This is possible because the set of balls $B(x,r)$ has VC dimension at most $d+2$.

We now bound $\mathbb{E}_{S\sim\mathscr{D}^n}[\sup_{x\in\mathscr{X}} \sum_1^n w_i^S(x) 1_{\rho(x,x_i)>r_p(x)}]$ by dividing into cases where S satisfies and doesn't satisfy equation A.5.

Suppose S satisfies equation A.5. By condition 1. of Corollary 33, K is decreasing, and by Lemma 57, $r_p^\varepsilon(x) \geq c_p^\varepsilon r_p(x)$. Therefore, we have that for any $x \in \mathscr{X}$,

$$\begin{aligned}\sum_1^n K(\rho(x,x_i)/h_n) 1_{\rho(x,x_i)\geq r_p^\varepsilon(x)} &\leq \sum_1^n K(c_p^\varepsilon r_p(x)/h_n) \\ &\leq n\delta K(r_p(x)/h_n)),\end{aligned}$$

where the second inequality comes from equation A.4.

Next, by the definition of $r_p^\varepsilon(x)$, we have that $\mu(B(x,r_p^\varepsilon(x)) - \mu(B(x,r_p(x))) \leq \varepsilon$. Therefore, by applying equation A.5 two times, we see that for any $x \in \mathscr{X}$

$$\sum_1^n K(\rho(x,x_i)/h_n) 1_{r_p(x)<\rho(x,x_i)\leq r_p^\varepsilon(x)} \leq 3n\varepsilon K(r_p(x)/h_n).$$

Finally, we have that

$$\sum_1^n w_i^S(x) \geq \sum_1^n K(r_p(x)/h_n) 1_{\rho(x,x_i)\leq r_p(x)} \geq n(p-\varepsilon)K(r_p(x)/h_n).$$

Therefore, using all three of our inequalities, we have that for any $x \in \mathscr{X}$

$$\begin{aligned}
\sum_1^n w_i^S(x) 1_{\rho(x,x_i)>r_p(x)} &= \sum_1^n w_i^S(x) 1_{\rho(x,x_i)>r_p^\varepsilon(x)} + \sum_1^n w_i^S(x) 1_{r_p^\varepsilon \geq \rho(x,x_i)>r_p(x)} \\
&= \frac{\sum_1^n K(\rho(x,x_i)/h_n) 1_{\rho(x,x_i)>r_p^\varepsilon(x)} + \sum_1^n K(\rho(x,x_i)/h_n) 1_{r_p^\varepsilon \geq \rho(x,x_i)>r_p(x)}}{\sum_1^n K(\rho(x,x_i)/h_n)} \\
&\leq \frac{n\delta K(r_p(x)/h_n)) + 3n\varepsilon K(r_p(x)/h_n)}{n(p-\varepsilon)K(r_p(x)/h_n).} \\
&= \frac{\delta + 3\varepsilon}{p-\varepsilon}.
\end{aligned}$$

If S does *not* satisfy equation A.5, then we simply have $\sup_{x\in\mathscr{X}} \sum_1^n w_i^S(x) 1_{\rho(x,x_i)>r_p(x)} \leq 1$. Combining all of this, we have that

$$E_{S\sim\mathscr{D}^n} \sum_1^n w_i^S(x) 1_{\rho(x,x_i)>r_p(x)} \leq \delta(1) + (1-\delta)\frac{\delta+3\varepsilon}{p-\varepsilon}.$$

Since δ, ε can be made arbitrarily small, the result follows. □

By assumption, $\mathscr{X}$ is compact and therefore has diameter $D < \infty$. Define

$$t_n = \sqrt{n \log n K(\frac{D}{h_n})} \text{ for } 1 \leq n < \infty.$$

Lemma 59. $\lim_{n\to\infty} E_{S\sim D^n}[t_n \sup_{x\in\mathscr{X}} w_i^S(x)] = 0.$

Proof. Because K is a decreasing function, we have that $K(D/h_n) \leq K(\rho(x,x_i)/h_n) \leq K(0)$. As

a result, we have that for any $x \in \mathscr{X}$,

$$\begin{aligned} t_n \sup_{1 \le i \le n} w_i^S(x) &= \frac{t_n \sup_{1 \le i \le n} K(\rho(x,x_i)/h_n)}{\sum_1^n K(\rho(x,x_i)/h_n)} \\ &\le \frac{t_n K(0)}{nK(D/h_n)} \\ &= K(0)\sqrt{\frac{n \log nK(D/h_n)}{n^2 K(D/h_n)^2}} \\ &= K(0)\sqrt{\frac{\log n}{nK(D/h_n)}}. \end{aligned}$$

However, by condition 4. of Corollary 33, $\lim_{n\to\infty} \frac{n}{\log n} K(D/h_n) = \infty$. Therefore, since the above inequality holds for all $x \in \mathscr{X}$, we have that

$$\lim_{n\to\infty} E_{S\sim D^n}[t_n \sup_{x \in \mathscr{X}} w_i^S(x)] \le \lim_{n\to\infty} K(0)\sqrt{\frac{\log n}{nK(D/h_n)}} = 0.$$

□

Lemma 60. $\lim_{n\to\infty} E_{S\sim D^n} \frac{\log T(W,S)}{t_n} = 0.$

Proof. For $S \sim \mathscr{D}^n$, recall that $T(W,S)$ was defined as

$$T(W,S)|\{W_{x,\alpha,\beta} : x \in \mathscr{X}, 0 \le \alpha, 0 \le \beta \le 1\}|,$$

where $W_{x,\alpha,\beta}$ denotes

$$W_{x,\alpha,\beta} = \{i : \rho(x,x_i) \le \alpha, w_i^S(x) \ge \beta\}.$$

Our goal will to be upper bound $\log T(W,S)$.

The key observation is that $W_{x,\alpha,\beta}$ is precisely the set of x_i for which $\rho(x,x_i) \le r$ where r is some threshold. This is because the restriction that $w_i^S(x) \ge \beta$ can be directly translated into $\rho(x,x_i) \le r$ for some value of r, as K is a monotonically decreasing function. Thus, $T(W,S)$ is the number of subsets of S that can be obtained by considering the interior of some ball $B(x,r)$

centered at x with radius r.

We now observe that the set of closed balls in $\mathbb{R}^d$ has VC-dimension at most $d+2$. Thus by Sauer's lemma, there are at most $O(n^{d+2}$ subsets of $\{x_1, x_2, \ldots, x_n\}$ that can be obtained from closed balls. Thus $T(W,S) \leq O(n^{d+2}$.

Finally, we see that

$$\lim_{n\to\infty} \frac{\log T(W,S)}{t_n} = \lim_{n\to\infty} \frac{O(d\log n)}{\sqrt{n\log nK(\frac{D}{h_n})}} \leq \lim_{n\to\infty} \sqrt{\frac{O(d\log n)}{nK(\frac{D}{h_n})}} = 0,$$

with the last equality holding by condition 4. of Corollary 33. □

Finally, we note that Corollary 33 is an immediate consequences of Lemmas 55, 58, 59, and 60, as we can simply apply Theorem 31.

A.3 Useful Technical Definitions and Lemmas

Lemma 61. *Let μ be a measure over $\mathbb{R}^d$, and let $\mathscr{A}$ denote a countable collections of measurable sets A_i such that $\mu(\bigcup_{A\in\mathscr{A}} A) < \infty$. Then for all $\varepsilon > 0$, there exists a finite subset of $\mathscr{A}$, $\{A_1, \ldots, A_m\}$ such that*

$$\mu(A_1 \cup A_2 \cup \cdots \cup A_m) > \mu(\bigcup_{A\in\mathscr{A}} A) - \varepsilon.$$

Proof. Follows directly from the definition of a measure. □

A.3.1 The support of a distribution

Let μ be a probability measure over $\mathbb{R}^d$.

Definition 62. *The **support** of μ, $supp(\mu)$, is defined as all $x \in \mathbb{R}^d$ such that for all $r > 0$, $\mu(B(x,r)) > 0$.*

From this definition, we can show that $supp(\mu)$ is closed.

Lemma 63. *$supp(\mu)$ is closed.*

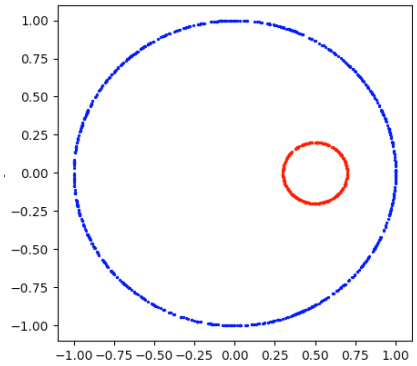


Figure A.1. Our data distribution $\mathscr{D} = (\mu, \eta)$ with μ^+ shown in blue and μ^- shown in red. Observe that this simple distribution captures varying distances between the red and blue regions, which necessitates having varying sizes for robustness regions.

Proof. Let x be a point such that $B(x,r) \cap supp(\mu) \neq \emptyset$ for all $r > 0$. It suffices to show that $x \in supp(\mu)$, as this will imply closure.

Let x be such a point, and fix $r > 0$. Then there exists $x' \in B(x, r/2)$ such that $x' \in supp(\mu)$. By definition, we see that $\mu(B(x', r/3)) > 0$. However, $B(x', r/3) \subset B(x,r)$ by the triangle inequality. it follows that $\mu(B(x,r)) > 0$. Since r was arbitrary, it follows that $x \in supp(\mu)$. □

A.4 Experiment Details

Data Distribution

Our data distribution $\mathscr{D} = (\mu, \eta)$ is over $\mathbb{R}^2 \times \{\pm 1\}$, and is defined as follows. We let μ^+ consist of a uniform distribution over the circle $x^2 + y^2 = 1$, and μ^- consist of the uniform distribution over the circle $(x-0.5)^2 + y^2 = 0.04$. The two distributions are weighted so that we draw a point from μ^+ with probability 0.7, and μ^- with probability 0.3. Finally, we utilize label noise 0.2 meaning that the label y matches that given by the Bayes optimal with probability 0.2. In summary, $\mathscr{D}$ can be described with the following 4 cases:

1. With probability 0.7×0.8, we select (x,y) with $x \in \mu^+$ and $y = +1$.

2. With probability 0.7×0.2, we select (x,y) with $x \in \mu^+$ and $y = -1$.

3. With probability 0.3×0.8, we select (x, y) with $x \in \mu^-$ and $y = -1$.

4. With probability 0.3×0.2, we select (x, y) with $x \in \mu^-$ and $y = +1$.

We also include a drawing (Figure A.1) of the support of $\mathscr{D}$, with the positive portion μ^+ shown in blue and the negative portion, μ^- shown in red.

Computing Robustness Regions

Recall that in order to measure robustness, we utilize the so-called partial neighborhood preserving regions V_x^κ (Definition 26) for varying values of κ. In the case of our data distribution $\mathscr{D}$, V_x^κ consists of points closer to x by a factor of κ than they are to μ^- (resp. μ^+) when $x \in \mu^+$ (resp. μ^-). To represent a region V_x^κ, we simply use a function f that verifies whether a given point $x' \in V_x^\kappa$. While this methodology is not sufficient for training general classifiers (for a whole litany of reasons: to begin with it assumes full knowledge of the distribution), it will suffice for our toy synthetic experiments.

Trained Classifiers

We train two classifiers, both of which are kernel classifiers.

The first classifier is an exponential kernel classifier with bandwidth function $h_n = \frac{1}{10\sqrt{\log n}}$ and kernel function $K(x) = e^{-x}$.

The second classifier is a polynomial kernel classifier with bandwidth function $h_n = \frac{1}{10n^{1/3}}$ and kernel function $K(x) = \frac{1}{1+x^2}$.

Both of these kernels are regular kernels, and both bandwidths satisfy sufficient conditions for consistency with respect to accuracy. In other words, both of these classifiers will converge towards the accuracy of the Bayes optimal.

However, the first classifier is selected to satisfy the criterion of Corollary 33, whereas the second is not. This distinction is reflected in our experiments.

Verifying Robustness

To verify the robustness of classifier f at point x (with respect to V_x^K), we simply do a grid search with grid parameter 0.01. We grid the entire regions into points with distance at most 0.01 between them, and then verify that f has the desired value at all of those points. To ensure proper robustness, we also simply verify that f cannot change enough within a distance of 0.01 by constructing an upper bound on how much f can possibly change. For kernel classifiers, this is simple to do as there is a relatively straightforward upper bound on the gradient of a Kernel classifier.

Bibliography

[1] 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, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

[2] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples, March 20 2014. URL http://arxiv.org/abs/1412.6572.

[3] Daniel Lowd and Christopher Meek. Adversarial learning. In Robert Grossman, Roberto J. Bayardo, and Kristin P. Bennett, editors, *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Chicago, Illinois, USA, August 21-24, 2005*, pages 641–647. ACM, 2005. ISBN 1-59593-135-X.

[4] Ludwig Schmidt, Shibani Santurkar, Dimitris Tsipras, Kunal Talwar, and Aleksander Madry. Adversarially robust generalization requires more data. In *Advances in Neural Information Processing Systems 31*, pages 5014–5026. Curran Associates, Inc., 2018.

[5] 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 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, pages 5120–5129, 2018.

[6] Omar Montasser, Steve Hanneke, and Nathan Srebro. VC classes are adversarially robustly learnable, but only improperly. In *Conference on Learning Theory, COLT 2019, 25-28 June 2019, Phoenix, AZ, USA*, pages 2512–2530, 2019.

[7] Hadi Salman, Greg Yang, Jerry Li, Pengchuan Zhang, Huan Zhang, Ilya P. Razenshteyn, and Sébastien Bubeck. Provably robust deep learning via adversarially trained smoothed classifiers. *CoRR*, abs/1906.04584, 2019. URL http://arxiv.org/abs/1906.04584.

[8] Dmitrii Avdiukhin, Slobodan Mitrovic, Grigory Yaroslavtsev, and Samson Zhou. Adversarially robust submodular maximization under knapsack constraints. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019.*, pages 148–156, 2019. doi:

10.1145/3292500.3330911.

[9] Luc Devroye, László Györfi, and Gábor Lugosi. *A Probabilistic Theory of Pattern Recognition*, volume 31 of *Stochastic Modelling and Applied Probability*. Springer, 1996.

[10] Charles Stone. Consistent nonparametric regression. *The Annals of Statistics*, 5(4):595–645, 1977.

[11] 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. URL http://arxiv.org/abs/1906.03310.

[12] Nicholas Carlini and David A. Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017*, pages 39–57, 2017.

[13] Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song. Delving into transferable adversarial examples and black-box attacks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017.

[14] Nicolas Papernot, Patrick D. McDaniel, Ian J. Goodfellow, Somesh Jha, Z. Berkay Celik, and Ananthram Swami. Practical black-box attacks against deep learning systems using adversarial examples. *ASIACCS*, 2017.

[15] Nicolas Papernot, Patrick D. McDaniel, Somesh Jha, Matt Fredrikson, Z. Berkay Celik, and Ananthram Swami. The limitations of deep learning in adversarial settings. In *IEEE European Symposium on Security and Privacy, EuroS&P 2016, Saarbrücken, Germany, March 21-24, 2016*, pages 372–387, 2016.

[16] Matthias Hein and Maksym Andriushchenko. Formal guarantees on the robustness of a classifier against adversarial manipulation. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 2266–2276. Curran Associates, Inc., 2017.

[17] Guy Katz, Clark W. Barrett, David L. Dill, Kyle Julian, and Mykel J. Kochenderfer. Towards proving the adversarial robustness of deep neural networks. In *Proceedings First Workshop on Formal Verification of Autonomous Vehicles, FVAV@iFM 2017, Turin, Italy, 19th September 2017.*, pages 19–26, 2017.

[18] 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, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, 2018.

[19] Nicolas Papernot, Patrick D. McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. Distillation as a defense to adversarial perturbations against deep neural networks. In *IEEE Symposium on Security and Privacy, SP 2016, San Jose, CA, USA, May 22-26, 2016*, pages 582–597, 2016.

[20] Aditi Raghunathan, Jacob Steinhardt, and Percy Liang. Certified defenses against adversarial examples. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, 2018.

[21] Aman Sinha, Hongseok Namkoong, and John C. Duchi. Certifying some distributional robustness with principled adversarial training. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, 2018.

[22] Laurent Amsaleg, James Bailey, Dominique Barbe, Sarah M. Erfani, Michael E. Houle, Vinh Nguyen, and Milos Radovanovic. The vulnerability of learning to adversarial perturbation increases with intrinsic dimensionality. In *2017 IEEE Workshop on Information Forensics and Security, WIFS 2017, Rennes, France, December 4-7, 2017*, pages 1–6, 2017.

[23] Chawin Sitawarin and David A. Wagner. On the robustness of deep k-nearest neighbors. In *2019 IEEE Security and Privacy Workshops, SP Workshops 2019, San Francisco, CA, USA, May 19-23, 2019*, pages 1–7, 2019.

[24] Maksym Andriushchenko and Matthias Hein. Provably robust boosted decision stumps and trees against adversarial attacks. In H. Wallach, H. Larochelle, A. Beygelzimer, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 12997–13008. Curran Associates, Inc., 2019.

[25] Alex Kantchelian, J. D. Tygar, and Anthony D. Joseph. Evasion and hardening of tree ensemble classifiers. *CoRR*, abs/1509.07892, 2015. URL http://arxiv.org/abs/1509.07892.

[26] Hongge Chen, Huan Zhang, Duane S. Boning, and Cho-Jui Hsieh. Robust decision trees against adversarial examples. *CoRR*, abs/1902.10660, 2019.

[27] Geoffrey W. Gates. The reduced nearest neighbor rule (corresp.). *IEEE Trans. Information Theory*, 18(3):431–433, 1972.

[28] Lee-Ad Gottlieb, Aryeh Kontorovich, and Pinhas Nisnevitch. Near-optimal sample compression for nearest neighbors. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 370–378, 2014.

[29] Peter E. Hart. The condensed nearest neighbor rule (corresp.). *IEEE Trans. Information Theory*, 14(3):515–516, 1968.

[30] Aryeh Kontorovich, Sivan Sabato, and Roi Weiss. Nearest-neighbor sample compression: Efficiency, consistency, infinite dimensions. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 1573–1583, 2017.

[31] Aryeh Kontorovich and Roi Weiss. A bayes consistent 1-nn classifier. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2015, San Diego, California, USA, May 9-12, 2015*, 2015.

[32] Steve Hanneke, Aryeh Kontorovich, Sivan Sabato, and Roi Weiss. Universal bayes consistency in metric spaces. *CoRR*, abs/1906.09855, 2019.

[33] Ingo Steinwart. Consistency of support vector machines and other regularized kernel classifiers. *IEEE Trans. Information Theory*, 51(1):128–142, 2005.

[34] Yao-Yuan Yang, Cyrus Rashtchian, Ruslan Salakhutdinov, and Kamalika Chaudhuri. A closer look at robustness vs. accuracy. In *Neural Information Processing Systems (NeuRIPS)*, 2020.

[35] Kamalika Chaudhuri and Sanjoy Dasgupta. Rates of convergence for nearest neighbor classification. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 3437–3445. Curran Associates, Inc., 2014.

[36] Minhao Cheng, Qi Lei, Pin-Yu Chen, Inderjit S. Dhillon, and Cho-Jui Hsieh. CAT: customized adversarial training for improved robustness. *CoRR*, abs/2002.06789, 2020. URL https://arxiv.org/abs/2002.06789.

[37] 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, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

[38] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Mach. Learn.*, 20(3): 273–297, 1995.

[39] Kristin P. Bennett and Erin J. Bredensteiner. Duality and geometry in SVM classifiers. In Pat Langley, editor, *Proceedings of the Seventeenth International Conference on Machine Learning (ICML 2000), Stanford University, Stanford, CA, USA, June 29 - July 2, 2000*, pages 57–64. Morgan Kaufmann, 2000.

[40] Yoav Freund and Robert E. Schapire. Large margin classification using the perceptron algorithm. *Mach. Learn.*, 37(3):277–296, 1999.

[41] Ulrike von Luxburg and Olivier Bousquet. Distance-based classification with lipschitz functions. In Bernhard Schölkopf and Manfred K. Warmuth, editors, *Computational Learning Theory and Kernel Machines, 16th Annual Conference on Computational Learning Theory and 7th Kernel Workshop, COLT/Kernel 2003, Washington, DC, USA, August 24-27, 2003, Proceedings*, volume 2777 of *Lecture Notes in Computer Science*, pages 314–328. Springer, 2003.

[42] Gamaleldin F. Elsayed, Dilip Krishnan, Hossein Mobahi, Kevin Regan, and Samy Bengio. Large margin deep networks for classification. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada*, pages 850–860, 2018.

[43] Kilian Q. Weinberger, John Blitzer, and Lawrence K. Saul. Distance metric learning for large margin nearest neighbor classification. In *Advances in Neural Information Processing Systems 18 [Neural Information Processing Systems, NIPS 2005, December 5-8, 2005, Vancouver, British Columbia, Canada]*, pages 1473–1480, 2005.

[44] Robi Bhattacharjee and Kamalika Chaudhuri. When are non-parametric methods robust? In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 832–841. PMLR, 2020.

[45] Sadia Chowdhury and Ruth Urner. On the (un-)avoidability of adversarial examples. *CoRR*, abs/2106.13326, 2021. URL https://arxiv.org/abs/2106.13326.

[46] Robert B. Ash. *Information theory*. Dover Publications, 1990.

[47] Sanjoy Dasgupta, Daniel J. Hsu, and Claire Monteleoni. A general agnostic active learning algorithm. In John C. Platt, Daphne Koller, Yoram Singer, and Sam T. Roweis, editors, *Advances in Neural Information Processing Systems 20, Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 3-6, 2007*, pages 353–360. Curran Associates, Inc., 2007.