Learning with non-Standard Supervision

by

Ruth Urner

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Computer Science

Waterloo, Ontario, Canada, 2013

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Machine learning has enjoyed astounding practical success in a wide range of applications in recent years—practical success that often hurries ahead of our theoretical understanding. The standard framework for machine learning theory assumes full supervision, that is, training data consists of correctly labeled *i.i.d.* examples from the same task that the learned classifier is supposed to be applied to. However, many practical applications successfully make use of the sheer abundance of data that is currently produced. Such data may not be labeled or may be collected from various sources.

The focus of this thesis is to provide theoretical analysis of machine learning regimes where the learner is given such (possibly large amounts) of non-perfect training data. In particular, we investigate the benefits and limitations of learning with unlabeled data in semi-supervised learning and active learning as well as benefits and limitations of learning from data that has been generated by a task that is different from the target task (domain adaptation learning). For all three settings, we propose Probabilistic Lipschitzness to model the relatedness between the labels and the underlying domain space, and we discuss our suggested notion by comparing it to other common data assumptions.

Acknowledgements

I would like to express my sincere gratitude to my supervisor, Wayne Oldford. While granting me freedom in my research, Wayne has been invaluably supportive, and has provided me with guidance and inspiration through many discussions. I also thank the other members of my committee, Sanjoy Dasgupta, Ming Li, Pascal Poupart and Mu Zhu. I greatly appreciate their support and their insightful comments on my work. Further, I am grateful to Naomi Nishimura for her support in the beginning of my studies at Waterloo.

I have spent several months doing research at the Hebrew University in Jerusalem and at ETH Zurich. I thank Joachim Buhmann for supporting my stay in Zurich. And I would like to extend a warm thank you to Sharon Wulff. I have found our working together fruitful, instructive and fun. I am very grateful to Shai Shalev-Shwartz for welcoming me into his research group in Jerusalem, for insightful discussions and for his support.

I would not be here without my family. I thank my parents for instilling confidence in my abilities and independence from early on. Moreover, Wilma has always given me her unconditional love and Michael the two most used items of these past years: My laptop and my bicycle. I thank my brother Max for being him and for leaving no doubt that he stands by me. Thanks also to Margit and Oma Marga for being affectionate and for visiting me in Canada.

My life would be less enjoyable without my friends and I am extremely happy about all of them. I send some special thanks to Colleen, for being my fellow female mathematician in CS grad school, to Robin for the singing and being awesome, to Jeanne for having visited me in (at least) five different countries by now, and to Silja and Philipp for finding the right words of reassurance when needed. Most importantly, I thank Axel for his companionship of so many years, and for all his care and encouragements towards my PhD endeavors.

I am deeply indebted to Shai Ben-David. Shai put trust in my capabilities when I had not much left. He has introduced me to learning theory, has taught me to do research and has over and over again impressed me with his sharpness, his profound understanding of the field, his instinct for exciting and relevant research questions, and his amazing mathematical intuition. There could not have been a better or more inspiring advisor. I am looking forward to all our future projects.

Table of Contents

Chapter 1

Introduction

Modern machine learning theory started about 30 years ago with the introduction of formal computational learning models by Valiant [1984]. In the past decade, machine learning has become one of the most influential tools of computer science, playing a key role in a very wide range of fields. It is extensively applied in bioinformatics, in finance, in astronomy, in social sciences as well as for many computer science tasks such as natural language processing and computer vision. However, with the expansion into practical use, new learning scenarios have emerged that are not well addressed by the standard machine learning theory.

Machine learning research comprises numerous tasks such as classification, regression, pattern recognition and clustering. Each of these tasks is investigated with various models for the input to the learner. These range from fully supervised to unsupervised and from batch to online or reinforcement learning models. In this work, we focus on binary classification in a batch model. Classification tasks occur in a variety of applications, including the analysis of medical data, spam detection, handwritten digit recognition, or sentiment analysis.

The standard theory for binary classification learning tasks models the environment as a probability distribution P over some domain $\mathcal{X}$ with labels in $\{0, 1\}$. The training data is assumed to be an *i.i.d.* labeled sample from this distribution P. A learning algorithm receives such a training sample and produces a *classifier*, that is, a function from the domain to the labels. The goal of the learner is to produce (with high probability) a classifier that has a low probability of misclassifying instances with respect to the distribution P.

This general framework is well understood. The theory of VC-dimension characterizes precisely for which classes of labeling functions (or classifiers) learning can succeed. However, there are many practical scenarios that are not well addressed by this standard framework. Many practical applications successfully make use of the sheer abundance of data that is currently produced. Such data may not be labeled or may be collected from divers sources.

An important issue, disregarded by the standard framework, is that obtaining correctly labeled training data for the task at hand can be expensive, if not impossible. The data might be noisy, obtaining labels for the training data might require hiring expensive experts or there might be labeled training data available only from a different (but related) task. Unlabeled training data is often easier and cheaper to come by than correctly labeled training data. In many practical applications unlabeled data has been successfully utilized to boost up learning algorithms. However, from a learning theory perspective, we are far from being able to explain this phenomenon.

The goal of the research in this thesis is to advance our understanding of the success of these heuristics. Which conditions on the underlying data distribution render the availability of unlabeled training data beneficial? When can the access to unlabeled data *provably* reduce the number of labeled instances required for learning? What kind of benefits can we get from unlabeled data?

In this thesis, we analyze three settings of learning with non-standard supervision: Semi-supervised learning, domain adaptation learning and active learning. The goal in all three of these learning scenarios is to save standard supervision (where by "standard supervision" we refer to the above described setting of learning with correctly labeled *i.i.d.* training examples with training and test distribution being identical). Semi-supervised learning approaches aim to save labeled examples by also considering (a large amount of) unlabeled training examples. In domain adaptation learning, there are no (or very little) labeled training examples from the relevant task available. To compensate for this, domain adaptation learners seek information from labeled training examples that were generated by a different task and also from unlabeled examples. Active learners get only an unlabeled sample as input, but are equipped with the ability to autonomously query for (some of the) labels. The goal of active learning is to succeed with as few as possible such label queries.

We focus on investigating the sample complexity of learning in these scenarios. That is, we provide upper and lower bounds on how many training examples a learner requires in order to output a low-error classifier. We focus particularly on labeled versus unlabeled sample complexity.

1.1 Contributions

Each of the learning settings that we investigate in this thesis, has been the subject of studies in learning theory before. We will now outline the original contributions that our work offers to these areas.

Probabilistic Lipschitzness

Several results in this thesis (in all three learning settings) employ the assumption of Probabilistic Lipschitzness (PL), a notion of clusterability of the learning task. This notion was introduced by Urner et al. [2011]. Prior to that, a very similar data parameter has been proposed by Steinwart and Scovel [2007] in the context of analyzing the performance of Support Vector Machines.

Probabilistic Lipschitzness is new to the analysis of the sample complexity of semi-supervised learning, active learning and domain adaptation learning. While data assumptions are essential in the analysis of these learning settings (this is implied by previously established lower bounds), PL, or any closely related parameter, has not been employed in this context before. We believe that PL provides a suitable way of modeling relatedness between the the distribution over the unlabeled instances and the labels. Such a relatedness is crucial to the possibility of gathering valuable information from unlabeled data for classification tasks. Thus, introducing PL to the analysis of learning settings that employ unlabeled data provides new insights into when and why unlabeled data is useful.

In addition to introducing and motivating the notion of Probabilistic Lipschitzness, we provide a discussion and comparisons of PL to other notions that have been proposed in the context of analyzing of the sample complexity of (standard and non-standard) learning.

Semi-Supervised Learning

In Semi-Supervised Learning (SSL), the learner gets as input a small sample of labeled examples and a large sample of unlabeled examples. Most previous work on semi-supervised learning focuses on investigating how the unlabeled data can be exploited to learn a classifier of lower error. Our work takes a different path: We investigate whether unlabeled data can be beneficial to learn classifiers that satisfy some application-defined requirement, such as being fast at prediction time or being easy to interpret.

We model this situation as *proper* semi-supervised learning, that is, SSL with the additional requirement that the output classifier is a member of a specific predefined hypothesis class. We propose an algorithmic framework for proper SSL, in which we first use the (small amount of) labeled data to learn some low-error classifier that is not necessarily from the predefined class. The algorithm then uses this classifier to label the (large) unlabeled input sample and uses this now labeled data to learn a classifier that complies with the properness requirements. This two-stage SSL learning has been employed in practice before by Liang et al. [2008] and Bucila et al. [2006]. However, it has not been subject to a thorough sample complexity analysis.

We analyze the performance of our algorithm under two data assumptions: realizability by a class and under Probabilistic Lipschitzness. We provide upper bounds on the labeled and unlabeled sample complexity in both these settings. We complement these with lower bounds on the sample complexity of these tasks without unlabeled data. The performance guarantees we prove for our algorithm together with the lower bounds imply that unlabeled data is provably beneficial for proper learning. To the best of our knowledge, provable reductions in label complexity with SSL have not been established before.

Domain Adaptation Learning

Domain Adaptation (DA) refers to the machine learning problem where the labeled training data was generated by a distribution (the source distribution) that is different from the one that the learned classifier should actually be applied to (the target distribution). However, unlabeled examples from that target distribution may be available. The formal framework for domain adaptation as learning from a labeled source sample together with an unlabeled target sample was introduced by Ben-David et al. [2006].

There have been various studies on the theory of domain adaptation learning in this framework. Most of these studies assume that source and target distribution are related with respect to a hypothesis class (via the $d_{\mathcal{A}}$-distance of Kifer et al. [2004] or the notion of discrepancy by Mansour et al. [2009]). Practical studies often make an assumption on the pointwise weight ratio between source and target density (Sugiyama et al. [2007]). We introduce a new measure to the analysis of domain adaptation, that generalizes the pointwise weight ratio. We propose to assume a bound on the weight ratio between source and target for a fixed class of subsets of the domain space. We also show that (under certain conditions on the class of subsets) this measure can be estimated from finite unlabeled source and target samples.

We investigate a popular DA paradigm under these assumptions: Reweighting a labeled source sample (to make it similar to a target generated sample) using unlabeled data from the target. This approach has been promoted by practical studies, in particular in combination with the pointwise weight ratio assumption, for example by Sugiyama et al. [2008]. Our contributions are two-fold: On the negative side, we prove that even under very strong assumptions about the relationship between source and target distribution and, in addition, a realizability assumption for the target task with respect to a very simple class, the required total sample sizes grow prohibitively. This shows that, in general, reweighting techniques require unrealistically large training samples. On the other hand, we present a reweighting algorithm, that, under slight additional assumptions, has a sample complexity that almost matches our lower bounds. Moreover, the algorithm shows that the (necessarily large) samples can be mostly unlabeled target samples. These results imply that unlabeled target generated data is provably beneficial for DA learning.

We also present an analysis of Nearest Neighbor learning under the assumption of Probabilistic Lipschitzness together with the bounded weight ratio. Nearest Neighbor approaches have not been analyzed for domain adaptation before.

Finally, we provide an analysis of proper DA learning. We present a lower bound that implies that target generated (possibly unlabeled) data is indispensable in this setting. Combining this with an approach for proper DA learning that is similar to our paradigm for proper SSL learning, we establish that unlabeled target samples are provably beneficial in this setting. The emphasis on proper learning, and the implications on the necessity and benefits of (unlabeled) target samples in this setting are new to the study of domain adaptation.

Active Learning

In Active Learning (AL), the learner receives an unlabeled sample and can autonomously query labels. The goal is to make as few as possible queries while still learning a low-error classifier. Active learning is applied in practice and there is also a rich body of literature on the theory of active learning. Most of these studies analyze active learning approaches that aim to use the queries in order to efficiently reduce a space of relevant hypotheses. Our work builds on ideas of Dasgupta and Hsu [2008], that take a different approach by analyzing active learning based on an assumption of clusterability of the data.

Dasgupta and Hsu [2008] propose a general framework for using active label queries when the learner is provided with a hierarichal clustering of the unlabeled data. We suggest a specific version of this framework and provide a new analysis of the obtained procedure under the assumption of Probabilistic Lipschitzness. The algorithm, PLAL, can be viewed as activising standard learners. It takes an unlabeled sample, queries the labels of some of its members, and outputs a full labeling of that sample. Assuming the data satisfies PL, we show that for several common learning paradigms, applying our procedure as a preprocessing leads to provable label complexity reductions (compared to any passive learning algorithm, under the same data assumptions).

1.2 Organization

Chapter 2 introduces the formal framework of standard machine learning theory. It also presents some well established results on the sample complexity of learning in this framework. These results serve as a baseline for our investigations of the sample complexity in non-standard learning settings. Chapter 3 introduces the notion of Probabilistic Lipschitzness. In addition to the formal definition we provide a discussion of this parameter that includes motivation and comparison to other data assumptions. The chapter also contains some new results on the sample complexity of standard learning under Probabilistic Lipschitzness. Our results on semi-supervised learning are presented in Chapter 4. Chapter 5 contains our study on domain adaptation and Chapter 6 our work on active learning. Each of these three latter chapters starts with an overview, provides a discussion of related work, and then proceeds to the formal presentation of the results. We end with some concluding remarks in Chapter 7.

The content of this thesis is based on four publications: The results on semi-supervised learning were published in "Unlabeled Data can Speed-up Prediction Time" by Urner et al. [2011]; the results on domain adaptation were published in "On the Hardness of Domain Adaptation and the Utility of Unlabeled Target Samples" by Ben-David and Urner [2012] and "Domain Adaptation–Can Quantity compensate for Quality?" by Ben-David and Urner [2013] (and the latter publication's earlier conference version Ben-David et al. [2012]); the results on active learning were published in "PLAL: Cluster-based active learning" by Urner et al. [2013].

Chapter 2

Definitions and Background

2.1 Overview

The focus of this thesis is to compare the number of labels required to learn (the *label complexity* or *labeled sample complexity*) in various learning frameworks. In particular, we compare the labeled sample complexity of semi-supervised learning, domain adaptation and active learning to the scenario, where a learner is provided only with a fully labeled *i.i.d.* sample from some training distribution and this distribution is identical to the distribution that the learned classifier will be applied to. In this work, we refer to this baseline setting as the *standard learning framework*.

We now introduce our formal model for standard learning. We provide formal notions of *learnability* and *sample complexity* (Section 2.2). We then briefly survey known results on the sample complexity of learning in this framework (Section 2.3). These results serve as a baseline for our exploration of the labeled sample complexity in other settings. Our model of learning is in line with the setting of (agnostic) PAC (Probably Approximately Correct) learning, a well established framework for analyzing binary classification tasks, first introduced by Valiant [1984].

2.2 The Standard Learning Framework

2.2.1 Basic notions

Mathematics notation We use common notation for sets and functions. In particular, for sets X and Y, we let Y^X denote the set of all functions from X to Y. To state that some object h is a function from X to Y, we also write $h : X \to Y$. For a function $h \in Y^X$ and a subset $U \subseteq X$, we let $h|_U$ denote the restriction of h to U, that is $h|_U : U \to Y$ and $h|_U(u) = h(u)$ for all $u \in U$.

If P is a probability distribution over some space $\mathcal{X}$ and $E \subseteq \mathcal{X}$ is an event, we use the notation $\Pr_{x \sim P}[x \in E]$ to denote $P(E)$. If $V : \mathcal{X} \to \mathbb{R}$ is a random variable, we denote its expectation of V by $\mathbb{E}_{x \sim P}[V(x)]$.

Learning theory notation We let $\mathcal{X}$ denote a *domain* set and $\mathcal{Y}$ some *label* set. In this work, we mostly consider $\mathcal{X} = [0,1]^d$ for some $d \in \mathbb{N}$ and always a binary label set $\mathcal{Y} = \{0,1\}$. Whenever the domain is $\mathcal{X} = [0,1]^d$, we consider it equipped with the Euclidean metric and denote the distance of two domain points by $\|x - y\|$. For $x \in \mathcal{X}$, we denote the *ball of radius r around x* by $B_r(x) := \{z \in \mathcal{X} \ : \ \|x - z\| \leq r\}$.

A *hypothesis* (or *label predictor* or *classifier*), is a binary function $h : \mathcal{X} \to \{0,1\}$, and a *hypothesis class H* is a set of hypotheses. We model a *learning task* as some distribution P over $\mathcal{X} \times \{0,1\}$ that generates data. We denote the marginal distribution of P over $\mathcal{X}$ by $P_{\mathcal{X}}$ and let $l : \mathcal{X} \to [0,1]$ denote the induced conditional label probability function, $l(x) = P(y = 1|x)$. We call l the *labeling function* or *labeling rule* of the distribution P. We say that the labeling function is *deterministic*, if $l(x) \in \{0,1\}$ for all $x \in \mathcal{X}$. Otherwise, we call the labeling function *probabilistic.*

For some function $h : \mathcal{X} \to \{0,1\}$ we define the *error* of h with respect to P as

$$\mathrm{Err}_P(h) = \Pr_{(x,y)\sim P}[y \neq h(x)].$$

For a class H of hypotheses on $\mathcal{X}$, we let the smallest error of a hypothesis $h \in H$ with respect to P be denoted by

$$\mathrm{opt}_P(H) := \inf_{h \in H} \mathrm{Err}_P(h).$$

We call $\mathrm{opt}_P(H)$ the *approximation error* of the hypothesis class H with respect to P. We let $\mathrm{opt}_P = \inf_{h \in \{0,1\}^{\mathcal{X}}} \mathrm{Err}_P(h)$ denote the smallest possible error of any classifier over $\mathcal{X}$ with respect to P. We refer to opt_P as the *Bayes optimal error* of P. The *Bayes optimal classifier* is defined as $h_{\mathrm{bayes}}(x) = 1$ if $l(x) \geq 1/2$ and $h_{\mathrm{bayes}}(x) = 0$ if $l(x) < 1/2$. The Bayes optimal classifier attains the minimum error opt_P. Note that, if the labeling function is deterministic, then $\mathrm{opt}_P = 0$.

Let $S = ((x_1, y_1), \ldots, (x_n, y_n)) \in (\mathcal{X} \times \{0,1\})^n$ be a finite sequence of labeled domain points. We define the *empirical error* of a hypothesis with respect to S as

$$\mathrm{Err}_S(h) = \frac{1}{|S|} \sum_{(x,y) \in S} |y - h(x)|.$$

A *standard learner* $\mathcal{A}$ is an algorithm that takes a sequence $S = ((x_1, y_1), \ldots, (x_n, y_n))$ and outputs a hypothesis $h : X \to \{0,1\}$. Formally,

$$\mathcal{A} : \textstyle\bigcup_{m=1}^{\infty} (\mathcal{X} \times \{0,1\})^m \to \{0,1\}^{\mathcal{X}}.$$

2.2.2 Learnability and sample complexity

We distinguish (and compare) three types of learning: *General learning*, *class learning* and *proper learning*. The three learning scenarios differ in the way the quality of the learned hypothesis is evaluated. Next, we provide formal definitions of learnability and sample complexity in these scenarios. As the learner receives only one sample, which is labeled, in the standard framework, the *sample complexity* of a standard learner is identical to its *label complexity* or *labeled sample complexity*. We start with general definitions of learnability and sample complexity and then explain how the three special types mentioned above correspond with these general definitions.

As a disclaimer we point out that characterizing learnability of a hypothesis class in terms of its VC dimension (see Section 2.3.2), requires some subtle measure theoretic conditions. A commonly assumed such condition is the *well behavedness* of a class introduced by Ben-David as part of a study by Blumer et al. [1989]. Virtually any hypothesis class that is considered in the context of machine-learning applications is well behaved. In this thesis, we therefore always implicitly assume the well-behavedness a hypothesis class.

Definition 1 (Learnability)**.** Let $\mathcal{X}$ denote some domain. We say that an algorithm $\mathcal{A}$ *learns* some class of binary classifiers $\mathcal{C} \subseteq \{0,1\}^{\mathcal{X}}$ with respect to a set of distributions $\mathcal{Q}$ over $\mathcal{X} \times \{0,1\}$, if there exists a function $m : (0,1) \times (0,1) \to \mathbb{N}$ such that, for all distributions $P \in \mathcal{Q}$, and for all $\epsilon > 0$ and $\delta > 0$, when given an *i.i.d.* sample of size at least $m(\epsilon, \delta)$ from P, then, with probability at least $1 - \delta$ over the sample, $\mathcal{A}$ outputs a classifier $h : X \to \{0,1\}$ with error at most $\mathrm{opt}_P(\mathcal{C}) + \epsilon$. In this case, for given ϵ and δ, we also say that the algorithm (ϵ, δ)*-learns* $\mathcal{C}$ from $m(\epsilon, \delta)$ examples with respect to $\mathcal{Q}$.

Let $\mathcal{F} \subset \mathbb{N}^{((0,1)\times(0,1))}$ be a class of functions from $(0,1) \times (0,1)$ to $\mathbb{N}$. We use the symbol $\leq$ for functions in $\mathcal{F}$ in the sense of being pointwise smaller, that is we write $m \leq n$ for $m, n \in \mathcal{F}$ if $m(x) \leq n(x)$ for all $x \in ((0,1) \times (0,1))$. The *pointwise minimum* over all functions in $\mathcal{F}$ is defined by $\min_{\mathcal{F}}(x) = \min\{f(x) \ : \ f \in \mathcal{F}\}$ for all $x \in (0,1) \times (0,1)$. Note that the pointwise minimum function over all functions m that satisfy the above learnability condition, also satisfies the condition. We use term “smallest function” in this sense in the below definition of sample complexity and, as argued, this is well defined in this context.

Definition 2 (Sample Complexity)**.** We call the smallest function $m : (0,1) \times (0,1) \to \mathbb{N}$ that satisfies the condition of Definition 1 the *sample complexity of the algorithm* $\mathcal{A}$ *for learning* $\mathcal{C}$ *with respect to* $\mathcal{Q}$. We denote this function by $m[\mathcal{A}, \mathcal{Q}, \mathcal{C}]$. We call the smallest function $m : (0,1) \times (0,1) \to \mathbb{N}$ such that there exists a learner $\mathcal{A}$ with $m[\mathcal{A}, \mathcal{Q}, \mathcal{C}] \leq m$ the *sample complexity of learning* $\mathcal{C}$ *with respect to* $\mathcal{Q}$ and denote this function by $m[\mathcal{Q}, \mathcal{C}]$. We omit $\mathcal{Q}$ in this notation, when $\mathcal{Q}$ is the set of all distributions over $\mathcal{X} \times \{0,1\}$, and call $m[\mathcal{C}]$ the *sample complexity of learning* $\mathcal{C}$.

General learning

We refer to learning as *general learning* if the class $\mathcal{C}$ considered in the above definition is the set of all functions from $\mathcal{X}$ to $\{0,1\}$. For general learning, the quality of the learned hypothesis is thus measured with respect to the Bayes optimal error of the data generating distribution.

Definition 3 (General Learning)**.** We say that an algorithm $\mathcal{A}$ *learns* with respect to a set of distributions $\mathcal{Q}$ over $\mathcal{X} \times \{0,1\}$ if it satisfies Definition 1 for $\mathcal{C} = \{0,1\}^{\mathcal{X}}$.

As $\mathcal{C}$ is the class of all binary classifiers, we omit it in our notation for the sample complexity of general learning. We denote the *sample complexity of algorithm* $\mathcal{A}$ for general learning with respect to $\mathcal{Q}$ by $m[\mathcal{A}, \mathcal{Q}]$ and the *sample complexity* of general learning with respect to $\mathcal{Q}$ by $m[\mathcal{Q}]$.

We employ this definition of learnability mainly in the context of Nearest Neighbor learning (see Definition 11).

Class learning (PAC learning)

For class learning, the quality of the learned hypothesis is measured with respect to the approximation error of a fixed hypothesis class H. In this situation, learnability is usually considered with respect to all distributions or with respect to all distributions that are realizable by a certain hypothesis class (see Definition 4 for realizability below). This is the setting called *PAC learning* (Valiant [1984]). However, in the non-standard learning frameworks (like semi-supervised learning or active learning) we investigate in this work, reductions of label complexity are often provably impossible when learnability is considered with respect to all distributions. Therefore, learning is investigated under specific conditions on the data generating distribution, that is, the class of distributions is restricted. For a fair comparison of label complexity bounds it is thus necessary to also consider standard learning under the same conditions. We therefore use Definition 1 as our notion of *class learnability* with respect to specific sets of distributions. We now explain how the framework of PAC learning relates to our definitions.

In the setting of PAC learning, we distinguish between learning in the *realizable case* and *agnostic* learning.

Definition 4 (Realizability)**.** A distribution P over $\mathcal{X} \times \{0,1\}$ is *realizable* by a hypothesis class $H \subseteq \{0,1\}^{\mathcal{X}}$ if there exist a classifier $h \in H$ such that $\mathrm{Err}_P(h) = 0$. We let $\mathcal{Q}_H^{\mathcal{X}}$ denote the class of all distribution over $\mathcal{X}$ that are realizable by H, and $\mathcal{Q}_H^d$ denote the class of all such distribution over $[0,1]^d$.

Definition 5 (PAC Learning)**.** We say that an algorithm $\mathcal{A}$ *agnostically PAC learns* some hypothesis class H over $\mathcal{X}$ if it satisfies Definition 1 with respect to the set of all distributions over $\mathcal{X} \times \{0,1\}$. In this case, we also call $\mathcal{A}$ an *agnostic (PAC) learner*. We say that

$\mathcal{A}$ *PAC learns* some hypothesis class H (in the realizable case) if $\mathcal{A}$ satisfies Definition 1 for learnability with respect to the set of distributions $\mathcal{Q}_H^{\mathcal{X}}$ and call $\mathcal{A}$ a *PAC learner* in this case.

In accordance with Definition 2, we denote the sample of complexity of agnostically PAC learning a class H by $m[H]$ and the sample complexity of PAC learning H in the realizable case by $m[\mathcal{Q}_H^{\mathcal{X}}, H]$.

Proper learning

For proper learning, the learner is required to output a hypothesis from a fixed hypothesis class H and, as in the class learning scenario, the quality of the output is measured with respect to the approximation error of H.

Definition 6 (Proper Learning)**.** We say that an algorithm $\mathcal{A}$ *properly learns* some hypothesis class H over $\mathcal{X}$ with respect to a set of distribution $\mathcal{Q}$ over $\mathcal{X} \times \{0,1\}$, if it satisfies Definition 1 and always outputs a classifier $h \in H$.

We denote by $m[\mathcal{A}, \mathcal{Q}, H, \text{proper}]$ the *sample complexity of an algorithm $\mathcal{A}$ for properly learning H with respect to $\mathcal{Q}$* and by $m[\mathcal{Q}, H, \text{proper}]$ *sample complexity of properly learning H with respect to $\mathcal{Q}$*. We omit $\mathcal{Q}$ in this notation, when $\mathcal{Q}$ is the set of all distributions over $\mathcal{X} \times \{0,1\}$, and call $m[H, \text{proper}]$ the *sample complexity of properly learning H*.

2.2.3 Use of Landau notation

In this work, we investigate the sample complexity of algorithms and learning tasks as a function of $1/\epsilon$. Whenever we use Landau-notation to denote some functions' growth behavior, this function is considered as a function ϵ only, and we consider the asymptotic behavior as ϵ tends to 0.

2.3 Sample Complexity in Standard Framework

We now present some well established upper and lower bounds on the sample complexity of learning in the three settings presented above. These bounds are the baseline for our exploration of sample complexity bounds in non-standard learning settings. Most of these results can be found in Shalev-Shwartz and Ben-David [2014] and we provide a reference therein for each theorem; note however, that we rephrased them to comply with our notation.

2.3.1 General learning

We start with a negative result that is known as the *No-Free-Lunch principle*. It states that for every learning algorithm, there is a task that the algorithm fails on. In other words, there is no "universal learner".

Theorem 7 (No-Free-Lunch; Theorem 5.1 in Shalev-Shwartz and Ben-David [2014])**.** *Let $\mathcal{X}$ be a finite domain, let $\mathcal{A}$ be a learner and let $m \in \mathbb{N}$, $m \leq |\mathcal{X}|/2$ be a sample size. There exists a distribution P over $\mathcal{X} \times \{0, 1\}$ with a deterministic labeling function such that*

$$\Pr_{S \sim P^m}[\mathrm{Err}_P(\mathcal{A}(S)) \geq 1/8] \geq 1/7.$$

Note that the proof of this theorem provided in Shalev-Shwartz and Ben-David [2014] contains the statement that there exists a distribution P over $\mathcal{X} \times \{0, 1\}$ with a deterministic labeling function such that

$$\mathbb{E}_{S \sim P^m}[\mathrm{Err}_P(\mathcal{A}(S))] \geq 1/4.$$

We will also employ this version of the No-Free-Lunch statement.

The impossibility result stated in the No-Free-Lunch Theorem is a consequence of allowing all possible labeling functions in the class of distributions to be learned. Roughly speaking, if a learner gets to see the labels of less than half of the domain points, it can only guess the labels of the remaining points, and this guess will be wrong half of the time (leading to the expected error exceeding 1/4).

The No-Free-Lunch Theorem implies that there exists no algorithm that learns the class of all binary classifiers with respect to the class of all distributions over an infinite domain. One task of the theory of machine learning is thus, to formally model properties of “benignity” of learning tasks that enable learnability. Ideally, such properties should also comply with our intuition about “naturally occurring learning tasks. We then analyze general learning only with respect to the restricted classes of distributions that have these properties.

One restriction to avoid the No-Free-Lunch phenomenon is to consider only distributions whose labeling function satisfies a *Lipschitz condition* (this represents one possible way of modeling “benignity” of a distribution). Recall the definition of (standard) Lipschitzness:

Definition 8 (Lipschitz condition; Definition 12.6 in Shalev-Shwartz and Ben-David [2014])**.** A function $f : \mathbb{R}^d \to \mathbb{R}$ is *L-Lipschitz* if

$$|f(x) - f(y)| \;\leq\; L \cdot \|x - y\|$$

holds for all $x, y \in [0,1]^d$.

If the labeling function of a distribution is deterministic, and the domain is $\mathcal{X} = [0,1]^d$, then the Lipschitz condition enforces the labeling to be constant. To avoid this, we relax the requirement of Lipschitzness to only hold on sets of positive probability mass (as formalized in the next definition). We introduce the following notation for classes of distributions over $[0,1]^d \times \{0,1\}$ whose labeling function satisfies the Lipschitz condition:

Definition 9. (Classes of Lipschitz distributions; deterministic Lipschitz labeling functions) Let $\mathcal{X} = [0,1]^d$. We let $\mathcal{Q}_L^d$ denote the class of distributions over $\mathcal{X} \times \{0,1\}$ whose labeling function is L-Lipschitz. Further, we let $\mathcal{Q}_{L,\mathrm{det}}^d$ denote the class of all distributions P over $\mathcal{X} \times \{0,1\}$ with a deterministic labeling function l such that

$$\Pr_{x \sim P_{\mathcal{X}}, y \sim P_{\mathcal{X}}} [\; |l(x) - l(y)| \;>\; L \cdot \|x - y\| \;] \;=\; 0.$$

Slightly abusing the notions, we also refer to the labeling function of the distribution as being L-Lipschitz in this case .

The above situation of a deterministic labeling function being Lipschitz (with respect to a distribution) can also be viewed as classifying with a *margin* (where a Lipschitz-constant L corresponds to a margin of $1/2L$). As this term is more common for classifiers (e.g. for halfspaces), we now also formally introduce the notion of a margin-classifier.

Definition 10. (Margin classifier; Realizability with a margin) Let $\mathcal{X} \subseteq \mathbb{R}^d$ be some domain, P a distribution over $\mathcal{X} \times \{0,1\}$, and $h : \mathcal{X} \to \{0,1\}$ a binary classifier. We say that h is a *γ-margin classifier with respect to P* if

$$\Pr_{x \sim P_{\mathcal{X}}, y \sim P_{\mathcal{X}}} [\ h(x) \neq h(y) \ \wedge \ \|x - y\| \ < \ \gamma \ ] \ = \ 0.$$

If this holds for some function $h \in H$ for a hypothesis class $H \subseteq \{0,1\}^{\mathcal{X}}$, then we also say that P is *realizable by H with margin γ.*

The Nearest Neighbor (NN) algorithm is an example of a learner that learns the class of all distributions whose labeling function satisfies the Lipschitz-condition if the labeling is deterministic.

Definition 11 (Nearest Neighbor algorithm; Section 19.1 in Shalev-Shwartz and Ben–David [2014])**.** Given a sequence $S = ((x_1, y_1), \ldots (x_m, y_m))$, the *Nearest Neighbor algorithm* NN returns a classifier NN(S) that labels every domain point according to the label of its nearest sample point in S. Formally, for all $x \in [0,1]^d$:

$$\mathrm{NN}(S)(x) = y_i, \qquad \text{where} \qquad i = \mathrm{argmin}_{i \in \{1,\ldots,m\}} \|x - x_i\|.$$

For this algorithm, the following learning guarantee holds for the set of distributions with general (probabilistic) labeling rules:

Theorem 12 (Nearest Neighbor Learning; Theorem 19.3 in Shalev-Shwartz and Ben-David [2014])**.** *Let P a distribution in $\mathcal{Q}_L^d$. Then*

$$\mathbb{E}_{S \sim P^m}[\mathrm{Err}_P(\mathrm{NN}(S))] \ \leq \ 2 \ \mathrm{opt}_P \ + \ 4 \ L \ \sqrt{d} \ m^{-\frac{1}{d+1}}.$$

Note that this result does not comply with our definition of learnability as the error bound states convergence to *twice* the Bayes optimal error. This can be improved by classifying with (the average label of) k-Nearest Neighbors (see, for example, Theorem 19.5 in Shalev-Shwartz and Ben-David [2014]). In this work, we employ the 1-Nearest Neighbor algorithm for several results (Section 4.5 in Chapter 4 and Section 5.6 in Chapter 5). Thus, with Nearest Neighbor algorithm, we always refer to the 1-Nearest Neighbor algorithm. One could extend those results to using k-Nearest Neighbors and analogously improve the convergence behavior.

The above bound yields an $O(1/\epsilon^{d+1})$ bound of the sample size in order to guarantee excess error (the $(4L\sqrt{d}m^{-\frac{1}{d+1}})$-part of the bound) smaller than ϵ. In the case of deterministic labeling, we can obtain better upper bounds on the label complexity of general learning under the assumption of L-Lipschitzness, in that the dependence on ϵ is only linear. The following is a special case of Theorem 20, which we prove in Section 3.4.1.

Theorem 13. *The sample complexity of general learning the class of distributions $\mathcal{Q}^d_{L,\mathrm{det}}$ satisfies*

$$m[\mathcal{Q}^d_{L,\mathrm{det}}](\epsilon,\delta) \;\leq\; m[\mathrm{NN},\mathcal{Q}^d_{L,\mathrm{det}}](\epsilon,\delta) \;\leq\; \left(L\sqrt{d}\right)^d \frac{6}{\epsilon\delta \mathrm{e}}.$$

On the other hand, there are also lower bounds on the sample complexity of learning this class:

Theorem 14 (NN lower bound; Theorem 19.4 in Shalev-Shwartz and Ben-David [2014])**.** *For any $L > 1$, and every learner $\mathcal{A}$, there exists a distribution $P \in \mathcal{Q}^d_{L,\mathrm{det}}$, such that for sample sizes $m \;\leq\; \frac{(L+1)^d}{2}$ we have*

$$\Pr_{S\sim P^m}[\mathrm{Err}_P(\mathcal{A}(S)) \geq 1/8] \geq 1/7.$$

Thus, in summary, we get that the sample complexity of general learning the class of distributions $\mathcal{Q}^d_{L,\mathrm{det}}$ satisfies

$$\frac{(L+1)^d}{2} \;\leq\; m[\mathcal{Q}^d_{L,\mathrm{det}}](\epsilon,\delta) \;\leq\; \left(L\sqrt{d}\right)^d \frac{6}{\epsilon\delta \mathrm{e}}$$

for all $\epsilon \in (0, 1/8)$, $\delta \in (0, 1/7)$ and $L > 1$.

2.3.2 Class learning

The established framework to investigate learnability of binary hypothesis classes is PAC (Probably Approximately Correct) learning. The PAC learning framework considers learnability of a hypothesis class H with respect to *all distribution*. The No-Free-Lunch result (Theorem 7) implies that learnability in this sense for the hypothesis class of all binary classifiers is impossible. In the previous section, we have seen that restricting the class of distributions with respect to which we aim to learn, is a way of overcoming the pessimistic message of the NFL theorem. Another way to avoid the NFL phenomenon, is to restrict the hypothesis class, with respect to which we measure the performance of the output classifier.

We therefore focus our attention to classes of bounded *VC-dimension*. The VC-dimension is a measure of complexity of classes of binary classifiers that was introduced by Vapnik and Chervonenkis [1971]. It is well-known that it characterizes the PAC learnability of a hypothesis class. A class is learnable (in the sense of Definition 1 with respect to the class of all distributions) if and only if it has bounded VC-dimension (see Theorem 16 below).

Definition 15 (Shattering, VC-dimension; Definition 6.5 in Shalev-Shwartz and Ben–David [2014])**.** Let $\mathcal{X}$ be a domain and $H \subseteq \{0,1\}^{\mathcal{X}}$ hypothesis class over $\mathcal{X}$. We say that H *shatters* a subset S of $\mathcal{X}$ if the restriction of H to S is equal to the set of all binary functions with domain S. Formally,

$$\{h|_S \ : \ h \in H\} = \{0,1\}^S.$$

The *VC-dimension* of H is the maximal size of a subset of $\mathcal{X}$ that H shatters and is denoted as $\mathrm{VCdim}(H)$.

The following theorem provides upper and lower bounds on the sample complexity of class learning in the realizable and in the agnostic case. The upper bounds in these results are achieved by the ERM (Empirical Risk Minimization) principle. Given a labeled sample S from some learning task, an ERM algorithm for some hypothesis class H returns a classifier from H with smallest empirical error on the sample S. Recall that $\mathcal{Q}_H^{\mathcal{X}}$ denotes the class of all distributions over $\mathcal{X} \times \{0,1\}$ that are reliable by the class H.

Theorem 16 (Sample complexity of class learning; Theorem 5 in Shalev-Shwartz and Ben-David [2014])**.** *Let $\mathcal{X}$ be some domain, let H be a hypothesis class of functions from $\mathcal{X}$ to $\{0,1\}$ with $\mathrm{VCdim}(H) < \infty$. Then, there are absolute constants C_1, C_2 such that the following holds:*

Realizable case: *H is PAC learnable with sample complexity*

$$C_1 \frac{\mathrm{VCdim}(H) + \ln(1/\delta)}{\epsilon} \;\leq\; m[\mathcal{Q}_H^{\mathcal{X}}, H](\epsilon, \delta) \;\leq\; C_2 \frac{\mathrm{VCdim}(H)\ln(1/\epsilon) + \ln(1/\delta)}{\epsilon}$$

for all $\epsilon, \delta \in (0,1)$.

Agnostic case: *H is agnostically PAC learnable with sample complexity*

$$C_1 \frac{\mathrm{VCdim}(H) + \ln(1/\delta)}{\epsilon^2} \;\leq\; m[H](\epsilon, \delta) \;\leq\; C_2 \frac{\mathrm{VCdim}(H) + \ln(1/\delta)}{\epsilon^2}$$

for all $\epsilon, \delta \in (0,1)$.

2.3.3 Proper learning

A lower bound on the sample complexity of class learning is also a lower bound on the sample complexity of proper learning. As the upper bounds in Theorem 16 are realized by the ERM principle, they also hold for proper learning.

Corollary 17 (Sample complexity of proper learning)**.** *Let $\mathcal{X}$ be some domain, let H be a hypothesis class of functions from $\mathcal{X}$ to $\{0,1\}$ with $\mathrm{VCdim}(H) < \infty$. Then, with the constants C_1, C_2 from Theorem 16 the following holds:*

Realizable case: *H is properly learnable with sample complexity*

$$C_1 \frac{\mathrm{VCdim}(H) + \ln(1/\delta)}{\epsilon} \;\leq\; m[\mathcal{Q}_H^{\mathcal{X}}, H, \mathrm{proper}](\epsilon, \delta) \;\leq\; C_2 \frac{\mathrm{VCdim}(H)\ln(1/\epsilon) + \ln(1/\delta)}{\epsilon}$$

for all $\epsilon, \delta \in (0,1)$.

Agnostic case: *H is properly learnable with respect to all distributions with sample complexity*

$$C_1 \frac{\mathrm{VCdim}(H) + \ln(1/\delta)}{\epsilon^2} \;\leq\; m[H, \mathrm{proper}](\epsilon, \delta) \;\leq\; C_2 \frac{\mathrm{VCdim}(H) + \ln(1/\delta)}{\epsilon^2}$$

for all $\epsilon, \delta \in (0,1)$.

Chapter 3

Probabilistic Lipschitzness

3.1 Motivation

In this chapter we introduce the notion of *Probabilistic Lipschitzness (PL)*, a property that measures the of marginal-label relatedness of a distribution with respect to the geometry of the space. PL bounds the mass of points for which the labeling function changes significantly in a small area around these points. We employ this property mostly in the case of deterministic labeling. In this case, for every radius λ, PL bounds the mass of points, that lie in label-heterogeneous balls of radius λ by a function $\phi(\lambda)$.

Many common learning paradigms implicitly rely on the labeling function to comply with the geometry of the space, or, put more simply, they rely on close-by points being likely to have the same label (or conditional label probability). Obvious examples of such paradigms are Nearest Neighbor methods or algorithms that classify with halfspaces (or other geometrically defined classifiers). The success of such paradigms suggests that, for many label prediction tasks, there is a significant correlation between the geometry of the space, the marginal distribution over the data points and the labels. Under a suitable data representation, or feature choice, we expect that the closer two instances are, the less likely they are to have different labels. Probabilistic Lipschitzness is a measure that quantifies this correlation.

The notion of learnability introduced in the previous chapter considers a worst case scenario: In order to meet the success criteria, an algorithm has to perform well with respect to all possible data generating distributions. However, applications often perform better than what is suggested by the lower bounds in that framework of analysis. This is particularly the case for settings that exploit information from unlabeled data, such as semi-supervised learning or active learning. For both these learning regimes, there are lower bounds on the label complexity in the presence of unlabeled data that meet the complexity of learning in the standard regime (without access to unlabeled data). Despite these pessimistic prospects, many practical applications successfully boost learning performance by taking information from unlabeled data into account. A theory that aims to explain the benefits of unlabeled data for learning thus needs to identify properties, that are both realistic (in that we can expect real world tasks to comply with the property), and yet mathematically sufficiently accessible to allow for a sound analysis. PL aims to combine these requirements, by formally modeling an intuition that is implicit in many machine learning algorithm designs.

We now proceed to defining PL, and providing some examples of distributions and their Probabilistic Lipschitzness. In Section 3.3, we then discuss other data-assumptions that have been considered in previous studies, either in order to analyze standard learning not from a worst case perspective or in order to account for the utility of unlabeled data in non-standard learning regimes. Finally, in Section 3.4 we present some upper and lower bounds on the sample complexity of standard learning under PL.

Probabilistic Lipschitzness was introduced by Urner et al. [2011]. A very similar notion has been proposed earlier by Steinwart and Scovel [2007] (see discussion in Section 3.3.1 below).

3.2 Definition

We define Probabilistic Lipschitzness as a relaxation of standard Lipschitzness (see Definition 8). Loosely speaking, for PL, we require Lipschitzness to only hold with some (high) probability. That is, we allow the Lipschitz-condition to be violated for some fraction of the points. How large this fraction of points is allowed to be, is determined by the distance of the two points involved in the Lipschitz condition (larger distance implies more allowance).

While the standard Lipschitz condition can be readily applied to probabilistic labeling rules $l : X \to [0,1]$, it has strong implications in the case of deterministic labeling functions. A Lipschitz constant λ for a distribution with deterministic labeling function forces a $1/\lambda$ gap between differently labeled points. Thus, the standard Lipschitz condition for deterministic labeling functions implies that the data lies in label homogeneous regions (clusters) that are separated by $1/\lambda$-margins of weight zero with respect to the distribution. This is a rather strong assumption of label conform clusterability. PL weakens this assumptions by allowing the margins to "smoothen out". The relaxation from Lipschitzness to Probabilistic Lipschitzness is thus particularly relevant to the deterministic labeling regime. It allows to model the marginal-label relatedness without trivializing the setup.

Definition 18 (Probabilistic Lipschitzness)**.** Let $\mathcal{X}$ be some Euclidean domain and let $\phi : \mathbb{R} \to [0,1]$. We say that $f : \mathcal{X} \to \mathbb{R}$ is ϕ-Lipschitz with respect to a distribution $P_{\mathcal{X}}$ over $\mathcal{X}$ if, for all $\lambda > 0$:

$$\Pr_{x \sim P_{\mathcal{X}}} \left[\Pr_{y \sim P_{\mathcal{X}}} \left[\, |f(x) - f(y)| \; > \; 1/\lambda \, \|x - y\| \, \right] \; > \; 0 \right] \; \leq \; \phi(\lambda)$$

If, for some distribution $P = (P_{\mathcal{X}}, l)$, the labeling function l is ϕ-Lipschitz, then we also say that P satisfies the ϕ-Probabilistic Lipschitzness. We denote the set of all such distributions over $[0,1]^d \times \{0,1\}$ by $\mathcal{Q}_{\phi}^{d}$ and the set of all such distributions over $[0,1]^d \times \{0,1\}$ with deterministic labeling functions by $\mathcal{Q}_{\phi,\mathrm{det}}^{d}$. Given some PL-function ϕ and some ϵ, we let $\phi^{-1}(\epsilon)$ denote the smallest λ, such that $\phi(\lambda) \geq \epsilon$.

Note, that it would be straightforward to define PL for functions on any metric space. The above definition reflects that, in this work, we focus on Euclidean domains.

If a distribution P is ϕ-Lipschitz for some function ϕ, then there always exists a non-decreasing function $\phi' \leq \phi$ (pointwise) such that P is also ϕ'-Lipschitz. It is easy to see that the pointwise minimum function ϕ that satisfies Definition 18 for some distribution P is non-decreasing and satisfies $\phi(0) = 0$. For all PL-functions ϕ considered in this work, we will thus implicitly assume that ϕ is non-decreasing and satisfies $\phi(0) = 0$. We will also refer to that minimum function as the *Probabilistic Lipschitzness* of the P.

If a distribution $P = (P_{\mathcal{X}}, l)$ with a deterministic labeling function l is ϕ-Lipschitz, then the weight of points x that have a positive mass of points of opposite label in an λ-ball around them, is bounded by $\phi(\lambda)$. This definition generalizes the standard definition of Lipschitzness (Definition 8) in the following way: For points x and y at distance smaller than λ with opposite labels, the standard Lipschitz condition for Lipschitz constant $1/\lambda$ is violated as $|l(x) - l(y)| = 1 > 1/\lambda\|x - y\|$. Thus, if the labeling function l of a distribution is L-Lipschitz then it satisfies Probabilistic Lipschitzness with the function $\phi(\lambda) = 1$ if $\lambda \geq 1/L$ and $\phi(\lambda) = 0$ if $\lambda < 1/L$.

3.2.1 Examples of Probabilistic Lipschitzness

Linear Separators Let $P_{\mathcal{X}}$ be the uniform distribution over $\mathcal{X} = [0,1]^d$. If l is a linear separator, then $\phi(\lambda) = C \cdot \lambda$, for some constant C.

Ball Let $P_{\mathcal{X}}$ be the uniform distribution over $\mathcal{X} = [0,1]^d$. For some ball $B \subseteq \mathcal{X}$ we let l label points in B with label 1 and points outside B with label 0. Then the Probabilistic Lipschitzness is bounded by $\phi(\lambda) = C \cdot \lambda^d$ for some constant C.

Generalized Clusters To demonstrate how the marginal distribution influences the Probabilistic Lipschitzness, we consider distributions over $\mathcal{X} = [0,1]$ and let the labeling function l be 0 for $x \leq 1/2$ and 1 for $x > 1/2$. Now we let the density d of the distribution form clusters by setting $d(x) = c \cdot (x - 1/2)^\alpha$ for a suitable constant c (that ensures that d is a density function). Then we have for all $\lambda < 1/2$ that

$$\begin{aligned}
&\Pr_{x \sim P_{\mathcal{X}}}\left[\Pr_{y \sim P_{\mathcal{X}}} \left[\ |l(x) - l(y)| \ > \ 1/\lambda \,\|x - y\|\ \right] \ > \ 0 \right] \\
\leq \ & \int_{1/2-\lambda}^{1/2+\lambda} c \cdot (x - 1/2)^\alpha dx \\
\leq \ & 2 \cdot \int_0^\lambda c \cdot x^\alpha dx \ \leq \ C \cdot \lambda^{\alpha+1}
\end{aligned}$$

for some constant C. We will also refer to this type of PL function ($\phi(\lambda) = C \cdot \lambda^\alpha$) as *polynomial Lipschitzness*.

In the same manner, by letting the marginal distribution form clusters over the domain, we can construct an example of (what we call) *exponential Lipschitzness*, that is a distribution with PL function bounded by $\phi(\lambda) = C \cdot \mathrm{e}^{-1/\lambda}$ for some constant C. Again, we consider distributions over $\mathcal{X} = [0, 1]$ and let the labeling function l be 0 for $x \leq 1/2$ and 1 for $x > 1/2$. Let the density d of the distribution be $d(1/2) = 0$ and $d(x) = c\mathrm{e}^{-1/|1/2-x|}$ for $x \neq 1/2$, for a suitable constant c (that ensures that d is a density function). Now, we have for $\lambda \leq 1/2$ that

$$
\begin{aligned}
&\Pr_{x \sim P_{\mathcal{X}}} \left[\Pr_{y \sim P_{\mathcal{X}}} \left[\, |l(x) - l(y)| \; > \; 1/\lambda \, \|x - y\| \, \right] \; > \; 0 \right] \\
\leq \; & \int_{1/2-\lambda}^{1/2+\lambda} c\mathrm{e}^{-1/|1/2-x|} dx \\
= \; & 2 \cdot \int_0^\lambda c\mathrm{e}^{-1/x} dx \\
\leq \; & 2 \cdot \int_0^\lambda c \frac{1}{x^2} \mathrm{e}^{-1/x} dx \; \leq \; C \cdot \mathrm{e}^{-1/\lambda}
\end{aligned}
$$

for some constant C.

3.3 Comparison to Other Data Assumptions

We now discuss other data properties that have been investigated to account for various improvements of learning performance in comparison to the standard framework. Some studies show that under certain conditions the performance of standard learning improves. The *margin exponent* (Section 3.3.1) has been introduced in the context of studying the performance of SVMs. The *Tsybakov noise condition* (Section 3.3.2) was first employed to prove rates of the sample complexity of standard learning that interpolate between the realizable and the agnostic case. Other properties, that we discuss here, were introduced in the context of studying the benefits of unlabeled data. The *cluster assumption* (Section 3.3.4) is often invoked to motivate the design of algorithms that exploit unlabeled data in practice. *Niceness* and *Local conservativeness* (Section 3.3.3) and various notions of *compatibility of a distribution with a hypothesis class* (Section 3.3.5) were introduced in studies on the theory of learning with unlabeled data.

We loosely distinguish between *data assumptions* and *prior knowledge assumptions*. We refer to a condition or property as *prior knowledge*, if the property is used by the algorithm. We call a condition a *data assumption* if it is used only in the analysis of the algorithms performance. In this work, PL is only used as a data assumption.

Some of the properties we discuss now, are aimed to bound the "noise" of the labeling function of a distribution. The term noise is here used in the sense of the labeling function not being deterministic. In accordance with this, in the following discussions, a probabilistic labeling function $l : \mathcal{X} \to [0, 1]$ is considered "noisy" at points, where it is close to the value 1/2 (that is, both labels are assigned with high probability).

3.3.1 Margin exponent

Steinwart and Christmann [2008] propose a parameter called *margin exponent* in the context of analyzing performance of Support Vector Machines (SVMs). In the case of deterministic labeling functions, the margin exponent coincides with our Probabilistic Lipschitzness for PL-functions of the form $\phi(\lambda) = C \cdot \lambda^n$ (for some constant $C \in \mathbb{R}$ and some $n \in \mathbb{N}$).

Using our notation, the margin exponent is defined as follows: Let P be a distribution over $\mathcal{X} \times \{0, 1\}$, with labeling function $l : \mathcal{X} \to [0, 1]$. We define a function f_l for this distribution by

$$f_l(x) = \begin{cases} 0 & \text{if } l(x) < 1/2 \\ 1 & \text{if } l(x) > 1/2 \\ 1/2 & \text{if } l(x) = 1/2. \end{cases}$$

Now, the distribution has margin exponent α, if there exists a constant $c > 0$ such that

$$\Pr_{x \sim P_{\mathcal{X}}}[\exists\, y \in B_t(x) \;:\; |f_l(x) - f_l(y)| = 1] \;\leq\; c \cdot t^{\alpha}.$$

Like PL in the deterministic case, the above quantity bounds the mass of points that are close to the decision boundary and quantifies how fast this mass shrinks with the distance parameter t. The formulation of the margin exponent is sensitive to changing the values of the labeling function on sets of measure zero (as also pointed out by Steinwart and Christmann [2008]). Our formulation of PL avoids this sensitivity.

Chapter 8 of Steinwart and Christmann [2008] presents various examples of distributions with bounded margin exponent. These can equally serve as examples for a bound on the Probabilistic Lipschitzness of these distributions.

Steinwart and Christmann [2008] show that when the margin exponent is combined with a condition on the noise rate and marginal distribution near the decision boundaries of data, it can be used to bound the approximation error of Gaussian kernels. Intuitively, close to the decision boundary, the noise is required to be high and the density is required to be low. A version of this combination of margin exponent and bounds on the noise rate was earlier introduced by Steinwart and Scovel [2007] under the name *geometric noise exponent*.

3.3.2 Tsybakov noise condition

The *Tsybakov noise condition* was first introduced by Mammen and Tsybakov [1999] and is now a well known measure of how noisy a labeling function is. It controls the mass of points with labeling function l close to $1/2$, by bounding the rate at which this mass decreases as a function of the distance $|l(x) - 1/2|$.

The noise condition can be formally stated as follows: There exist parameters $C > 0$, $\alpha > 0$ and $t^* < 1/2$ such that

$$\Pr_{x \sim P_{\mathcal{X}}} [|l(x) - 1/2| \leq t] \;\leq\; C \cdot t^{\alpha}$$

for all $0 \leq t \leq t^*$. Mammen and Tsybakov [1999] use this condition in order to prove sample complexity rates that interpolate between the rates for class learning in the realizable case and the rates in the agnostic case. For these results, the Bayes optimal classifier is assumed to be a member of some known hypothesis class. The bounds further employ a measure of complexity for the hypothesis class in question (a measure different from the VC-dimension), which is argued to hold for some classes with smooth decision boundaries.

Tsybakov [2004] generalizes these results to the case where the Bayes classifier is only assumed to be a member of some collection of known hypothesis classes.

The above noise condition is weakest, if there exists a small α such that the inequality holds. At the other extreme, the strongest condition of the above type is bounding $l(x)$ away from $1/2$, that is $|l(x) - 1/2| > \eta$ for some $\eta > 0$ (with probability 1). This latter condition is also known as the *Massart condition*. Massart and Nédélec [2006] provide results on the sample complexity of learning under this condition.

The Massart condition has also been used to show benefits of active learning. Balcan et al. [2007] provide guarantees on the label complexity of active learning of halfspaces in the realizable case and under the Massart low noise condition when the Bayes optimal is a halfspace.

These low noise conditions are incomparable with our Probabilistic Lipschitz condition. While PL imposes smoothness on the labeling function (with respect to the underlying marginal), these conditions bound the labeling function away from the "noisy value" 1/2. In turn, the low noise conditions are typically employed to generalize results that were first obtained in the realizable setting. This is done by allowing noise that complies with the condition while still requiring the Bayes optimal classifier to a member of a fixed hypothesis class. PL, in contrast, becomes relevant for a different type of agnostic regime. While we mostly consider PL with deterministic labeling functions (that is with "zero noise" in the above low-noise framework), we do not impose any restrictions on the Bayes optimal (which is in our case identical to the deterministic labeling function).

3.3.3 Niceness and local conservativeness

Urner et al. [2012] investigate learnability from labels that are of low quality, for example as obtained via crowd-sourcing. This scenario is modeled as learning from two types of supervision: a *strong teacher* (domain expert) that provides labels according to the true labeling function; and a *weak teacher* (crowd) that labels according to an average of the labeling function in the neighborhood of a point. Urner et al. [2012] show that under some smoothness conditions on weak teachers' labels, learning is possible in this scenario from few expert labels (in comparison to the standard learning setting that considers only expert labels) and many weak labels. The smoothness conditions proposed in this work are referred to as *niceness* and *local conservativeness*.

Niceness formalizes the property of having not too many label-heterogeneous neighborhoods. Like PL, this can be viewed as stating that the underlying marginal distribution over the instances is sparse around the label decision boundaries. If the labeling function of the weak teacher is denoted by $\pi_s(x)$ (that is, π_s is the labeling function obtained by averaging the labels of points in a neighborhood according to some notion of similarity s), niceness is formally defined as follows:

For a function $\psi : [0, 1/2] \to [0, 1]$, we say that a distribution P is ψ-*nice* if for all $0 \le \lambda \le 1/2$

$$\Pr_{x \sim P_{\mathcal{X}}} [\min\{\pi_s(x), 1 - \pi_s(x)\} \ge \lambda] \le \psi(\lambda).$$

That is, the average label over a local neighborhoods is, for most of the instances, either close to 0 or close to 1. Note that this condition is similar to the Tsybakov low noise condition. However, the Tsybakov condition is a requirement on the labeling function l itself, while niceness imposes a similar requirement on the neighborhood-averages π_s. Further, in contrast to the noise bounding function in the Tsybakov condition, the niceness function ψ is not required to be of any particular type. If the neighborhood is a ball around the point, niceness can also be viewed as a generalization of Probabilistic Lipschitzness.

To prove learnability from weak teachers, Urner et al. [2012] impose a second condition on the weak teachers' labeling function. *Local conservativeness* bounds the probability of points that have a label different from the vast majority of their neighbors. Formally, for a function $\varphi : [0, 1] \to [0, 1]$, a distribution P is φ-*locally conservative* if for all $0 \le \lambda \le 1$.

$$\Pr_{x \sim P_{\mathcal{X}}} [|\pi_s(x) - l(x)| > 1 - \lambda] \le \varphi(\lambda).$$

That is, for most of the instances, the average label over their neighborhood is a good indication of the probability of having label 1 for that instance.

While these requirements can be viewed as relaxations of the Probabilistic Lipschitzness, the learnability results for this setting assume access to a large amount of weakly labeled data (that is a large amount of information about the average labeling around a point) and, in addition, the possibility to actively query the true labeling function.

3.3.4 Cluster assumption

Many studies on utilizing unlabeled data in practice invoke the *cluster assumption* or *low-density assumption* to motivate their methods. The cluster assumption implies that the data can be divided into clusters that are (almost) label-homogeneous and are separated by low-density regions. Label conform clusterability has been cited for justifying practical learning paradigms in all three learning settings considered in this work: semi-supervised learning, domain adaptation and active learning.

Most notably, semi-supervised algorithm designs are often motivated by stating the cluster assumption. Chapelle and Zien [2005] refer to this assumption when proposing a method for learning distances for transductive SVM. Rigollet [2007] proposes some formalization of the cluster assumption and presents a semi-supervised learning paradigm that relies on the label homogeneity of the clusters. Singh et al. [2008] analyze the use of unlabeled data in semi-supervised learning for a setup where the data is generated as label homogeneous clusters.

Shi and Sha [2012] propose a method for using unlabeled data from the target domain in a domain adaptation learning scenario. They motivate their approach with a setting where the label classes are separable and, in addition, where source and target clusters overlap. Thus, the label clusters are identifiable in source and target and it is possible to match the appropriate source and target groups. Patra and Bruzzone [2012] design an active learning scheme that relies on the data satisfying the cluster assumption.

Probabilistic Lipschitzness can be viewed as a way of formalizing the cluster assumption. PL implies that areas where both labels occur (that is the decision boundaries) can not be too heavy with respect to the underlying marginal distribution. PL thus provides one way of formally modeling and quantifying the cluster-assumption.

3.3.5 Compatibility with a class

It is interesting to note that, while practical studies often refer to some general notion of clusterability of the data, work on the theory of learning with unlabeled data mostly employs data assumptions that model some sort of compatibility of the data generating distribution with the hypothesis class to be learned.

Balcan and Blum [2005] propose a PAC-style framework for semi-supervised learning of a hypothesis class. For this, they introduce a notion of *compatibility* of a classifier with the unlabeled data distribution. This is defined in terms of a pointwise compatibility, that is a function $\chi : H \times \mathcal{X} \to [0,1]$. The compatibility of a hypothesis $h \in H$ is then defined as the expectation of this function and can be estimated from unlabeled data. An example for such a notion of compatibility for the class H of halfspaces is the margin (which can be estimated from the distances of unlabeled points to the halfspace). Balcan and Blum [2005] suggest to use unlabeled data to reduce the hypothesis space to only compatible classifiers and then use labeled data to learn a good classifier from this reduced hypothesis space. Under the assumption that the optimal (in terms of classification performance) classifier is compatible with the underlying marginal distribution, this framework admits finite sample learning bounds from both labeled and unlabeled data.

For domain adaptation, Ben-David et al. [2010a] analyze how well an ERM classifier for a labeled source sample performs respect to some target distribution. Obviously, without any further assumptions on the relatedness between the two tasks, the target-error of a source-ERM classifier can be arbitrarily high. To bound this error, Ben-David et al. [2010a] introduce two parameters that measure the relatedness between source and target task with respect to a hypothesis class H. The distance between the source and target distribution is measured by the d_H-distance, which is defined as the maximum difference between the source weight and the target weight of the symmetric difference between two classifiers in H. Formally, for a source distribution P^S and target distribution P^T,

$$d_H(P^S_{\mathcal{X}}, P^T_{\mathcal{X}}) = \sup_{h_1,h_2 \in H} |P^S_{\mathcal{X}}(h_1 \Delta h_2) - P^T_{\mathcal{X}}(h_1 \Delta h_2)|.$$

See also Section 5.4.2 in Chapter 5. The second parameter, λ_H, introduced in this work is the infimum, over all functions in H, of the sum of source and target error. The obtained error bound is then the sum of the usual error bound and these parameters. The success of this domain adaptation framework thus crucially depends on these parameters being small, that is, it depends on source and target being similar with respect to the class H that is to be learned. Kifer et al. [2004] argue that the d_H distance can be estimated from finite (unlabeled) samples. These notions and bounds have been generalized by Mansour et al. [2009] to real-valued function classes and more general loss functions. Here as well, the class H is assumed to serve the purpose of bridging between the source and target distribution.

The most prominent parameter that has been employed to analyze the performance active learning is the *disagreement coefficient*. It was introduced by Hanneke [2007]. For a hypothesis class H, it bounds the disagreement regions around the optimal classifier. More precisely, let h^* denote a classifier in H of minimum error. The disagreement coefficient of a class H with respect to a distribution $P_{\mathcal{X}}$ over a domain $\mathcal{X}$ of diameter D, is

$$\sup_{r \in [0,D]} \frac{P_{\mathcal{X}}(\bigcup_{h \in H : P_{\mathcal{X}}(h \Delta h^*) \leq r} (h \Delta h^*))}{r}.$$

For example, the disagreement coefficient of the class of initial segments of the unit interval under the uniform distribution over the unit interval is 2. The disagreement coefficient measures "into how many dimensions" the disagreement regions $h\Delta h^*$ spread. This becomes relevant for active learning strategies that aim to employ label queries to reduce the hypothesis space to contain only classifiers that are close to the optimal classifier h^*. The disagreement coefficient has been used in various studies to provide performance guarantees for active label querying strategies (Dasgupta et al. [2008], Beygelzimer et al. [2010], Beygelzimer et al. [2009]).

Probabilistic Lipschitzness differs from these assumptions in that its definition does not involve any specific hypothesis class. It is a property of the data generating distribution only. Results obtained under this assumption thus typically hold for learning any (learnable) hypothesis class. In addition, while the above mentioned properties can be viewed as prior knowledge assumptions, we use PL only as a data assumption

3.4 Learning under Probabilistic Lipschitzness

In this section, we prove upper and lower bounds on the sample complexity of learning under Probabilistic Lipschitzness of the underlying distribution.

3.4.1 Upper bounds

We start with providing upper bounds for general learning. These bounds are obtained by generalizing the sample complexity bounds of the Nearest Neighbor algorithm under standard Lipschitzness to the case of Probabilistic Lipschitzness.

For the upper bounds on the error probability of the Nearest Neighbor algorithm, we need the following technical lemma to bound the probability of points that do not have a close neighbor in the sample set S. This lemma and its proof can be found in Shalev-Shwartz and Ben-David [2014].

Lemma 19 (Lemma 19.2 in Shalev-Shwartz and Ben-David [2014]). *Let $C_1, C_2, \ldots, C_r$ be a sequence of subsets of some domain set $\mathcal{X}$ and let S be a set of points of size m, sampled i.i.d. according to some distribution $P_{\mathcal{X}}$ over $\mathcal{X}$. Then we have*

$$\mathbb{E}_{S\sim(P_{\mathcal{X}})^m}\left[\sum_{i:C_i\cap S=\emptyset} P_{\mathcal{X}}[C_i]\right] \leq \frac{r}{m\mathrm{e}}.$$

We start with bounding the sample complexity of Nearest Neighbor learning with respect to the class of distributions that have deterministic labeling functions and satisfy PL. Recall that $\mathcal{Q}^d_{\phi,\mathrm{det}}$ denotes the class of distributions over $[0,1]^d$ that satisfy the ϕ-PL assumption and have a deterministic labeling function.

Theorem 20. *Let $\phi : \mathbb{R} \to [0,1]$. The sample complexity of the Nearest Neighbor algorithm with respect to the class $\mathcal{Q}^d_{\phi,\mathrm{det}}$, is bounded by*

$$m[\mathrm{NN}, \mathcal{Q}^d_{\phi,\mathrm{det}}](\epsilon,\delta) \leq \frac{2}{\epsilon\,\delta\,\mathrm{e}}\left(\frac{\sqrt{d}}{\phi^{-1}(\epsilon/2)}\right)^d.$$

Proof. Let $\lambda = \phi^{-1}(\epsilon)$. Note that we can cover $\mathcal{X} = [0,1]^d$ with $r = \left(\sqrt{d}/\lambda\right)^d$ boxes (axis-alligned rectangles) $C_1, C_2, \ldots C_r$ of side-length $\lambda/\sqrt{d}$ and diameter λ. For a domain point $x \in \mathcal{X} = [0,1]^d$ we denote the box (from the above cover) that contains x by $C(x)$.

We can bound the error of the Nearest Neighbor classifier $\mathrm{NN}(S)$ for a sample S as follows:

$$\begin{aligned} \mathrm{Err}_P(\mathrm{NN}(S)) &= \Pr_{x\sim P_{\mathcal{X}}}[\mathrm{NN}(S)(x) \neq l(x)] \\ &\leq \Pr_{x\sim P_{\mathcal{X}}}[S_{\mathcal{X}} \cap C(x) = \emptyset] \;+\; \Pr_{x\sim P_{\mathcal{X}}}[\,\exists\, y \in C(x) \;:\; l(y) \neq l(x)], \end{aligned}$$

where $S_{\mathcal{X}}$ denotes the projection of S to $\mathcal{X}$ ($S_{\mathcal{X}}$ contains the sample points in S without their labels). That is, a point will only be labeled erroneously by $\mathrm{NN}(S)$, if it falls into a box that was not hit by the sample S or if it falls into a box that contains points of the other label.

By the choice of λ, the probability that a (test-)point falls into a box that contains sample points of the opposite label is bounded by ϵ.

We now show that for $m \geq \frac{1}{\epsilon\delta}\left(\frac{\sqrt{d}}{\phi^{-1}(\epsilon)}\right)^d$, the total mass of boxes that are not hit by a sample of size m is bounded by ϵ as well (with probability at least $(1-\delta)$ over the sample). This implies that, for such samples, the error of the Nearest Neighbor classifier is bounded by 2ϵ.

With Markov's inequality, Lemma 19 implies that for any $\epsilon > 0$ and m we have

$$\Pr_{S\sim P^m}\left[\left[\sum_{i:C_i\cap S=\emptyset} P[C_i]\right] > \epsilon\right] \;\leq\; \frac{r}{\epsilon m \mathrm{e}} \;=\; \frac{\left(\sqrt{d}/\lambda\right)^d}{\epsilon m \mathrm{e}}.$$

Setting this to be smaller than δ and solving for m now shows that a sample of size

$$m \;>\; \frac{\left(\sqrt{d}/\lambda\right)^d}{\epsilon\delta \mathrm{e}}$$

suffices to guarantee that with probability at least $(1-\delta)$, the error of the output function $\mathrm{NN}(S)$ is at most 2ϵ. Substituting $\epsilon/2$ for ϵ yields the statement in the theorem. □

We now provide an upper bound for the sample complexity of Nearest Neighbor learning for the case of general (possibly probabilistic) labeling functions. Recall that $\mathcal{Q}_\phi^d$ denotes the class of all distributions over $[0,1]^d \times \{0,1\}$ that satisfy the ϕ-PL assumption.

Theorem 21. *Let the domain be the unit cube, $\mathcal{X} = [0,1]^d$, let $\phi : \mathbb{R} \to [0,1]$. Then, for all $\lambda > 0$, for all $P \in \mathcal{Q}_\phi^d$,*

$$\mathbb{E}_{S\sim P^m}[\mathrm{Err}_P(\mathrm{NN}(S))] \leq 2\,\mathrm{opt}_P + \phi(\lambda) + 4\,\lambda^{-1}\sqrt{d}\,m^{-\frac{1}{d+1}}.$$

Proof. For $\lambda > 0$, we define the "non Lipschitz region" $U \subseteq \mathcal{X}$ of the domain as

$$U := \{x \in \mathcal{X} \;:\; \Pr_{z\sim P_{\mathcal{X}}}[|l(x) - l(z)| > 1/\lambda\|x-y\|] \;>\; 0\}.$$

By the definition of Probabilistic Lipschitzness, we have $P_{\mathcal{X}}(U) \leq \phi(\lambda)$. We have

$$\begin{aligned}
& \mathrm{Err}_P(\mathrm{NN}(S)) \\
= \;& \Pr_{(x,y)\sim P}(\mathrm{NN}(S)(x) \neq y) \\
\leq \;& P_{\mathcal{X}}(U) \Pr_{(x,y)\sim P}(\mathrm{NN}(S)(x) \neq y \mid x \in U) \;+\; P_{\mathcal{X}}(\mathcal{X}\setminus U) \Pr_{(x,y)\sim P}(\mathrm{NN}(S)(x) \neq y \mid x \in \mathcal{X}\setminus U) \\
\leq \;& \phi(\lambda) \;+\; 1 \cdot \Pr_{(x,y)\sim P}(\mathrm{NN}(S)(x) \neq y \mid x \in \mathcal{X}\setminus U).
\end{aligned}$$

For the term $\Pr_{(x,y)\sim P}(\mathrm{NN}(S)(x) \neq y \mid x \in \mathcal{X}\setminus U)$, we can employ the error bound in Theorem 12, and thereby obtain the claimed bound on the expected error.

This is also a special case of Theorem 52 that deals with the performance of the Nearest Neighbor algorithm in the context of domain adaptation. We provide a full proof of that Theorem in Chapter 5. □

3.4.2 Lower bounds

We now present lower bounds on the sample complexity of standard learning under the assumption of Probabilistic Lipschitzness. We start with a lower bound for general learning:

Theorem 22. *Let $\phi : \mathbb{R} \to [0,1]$ with $\phi(1) \geq 1$. For every passive learning algorithm $\mathcal{A}$ and every $\epsilon > 0$ there exists a distribution $P \in \mathcal{Q}^d_{\phi,\mathrm{det}}$ such that,*

$$m < \frac{d}{32\epsilon}\left(\frac{1}{\phi^{-1}(8\epsilon)}\right)^{d-1}$$

implies that

$$\mathbb{E}_{S\sim P^m}[\mathrm{Err}_P(\mathcal{A}(S))] > \epsilon.$$

Proof. The No-Free-Lunch theorem (Theorem 7) states that if a learner gets a sample of size less than half the size of the domain, then there is a distribution with a deterministic labeling function over this domain, such that the expected error of the learner for this sample size is at least 1/4.

We construct a distribution on $[0,1]^d$ that satisfies the ϕ-Lipschitzness as follows: We set $P(\bar{0}) = 1 - 8\epsilon$ and distribute the remaining mass of 8ϵ uniformly on points of a grid G of sidelength $\lambda = \phi^{-1}(8\epsilon)$ "at the far side of the surface of" $[0,1]^d$, i.e. the points $x = (x_1, \ldots x_d)$ where at least one of the x_i has value 1 and the others have values in $\{i\lambda \ : \ 1 \leq i \leq d\}$. Now P is ϕ-Lipschitz under any labeling of the grid points.

There are $|G| \geq d/(\lambda)^{d-1}$ such grid points. We show that with probability at least 1/2, a sample of size at most m hits less than $|G|/2$ gridpoints. The expected number of such hits is bounded by $8\epsilon m$, formally

$$\mathbb{E}_{S\sim P^m}[|S \cap G|] \ = \ 8\epsilon m.$$

Now Markov's inequality yields

$$\Pr_{S\sim P^m}[|S \cap G| > |G|/2] \ \leq \ \frac{16\epsilon m}{|G|}.$$

Now $m < \frac{d}{32\epsilon}(\frac{1}{\phi^{-1}(8\epsilon)})^{d-1}$ and $|G| \geq \frac{d}{(\lambda)^{d-1}} = \frac{d}{(\phi^{-1}(8\epsilon))^{d-1}}$ implies

$$\Pr_{S\sim P^m}[|S \cap G| > |G|/2] \ < \ \frac{16\epsilon \cdot \frac{d}{32\epsilon}(\frac{1}{\phi^{-1}(8\epsilon)})^{d-1}}{\frac{d}{(\phi^{-1}(8\epsilon))^{d-1}}} = \frac{1}{2}.$$

The above mentioned no-free-lunch result implies that, there is a labeling for the points on G, such that the learner $\mathcal{A}$ has expected error at least $\frac{1}{4} \cdot 8\epsilon = 2\epsilon$ given that the sample hits at most half of the grid points. Since we have shown that this happens with probability at least $1/2$ for samples of size at most m, the learners' expected error over all samples of size at most m is at least ϵ. □

We now provide a lower bound on the sample complexity of proper learning under Probabilistic Lipschitzness.

Theorem 23. *Let $\mathcal{X} = [0,1]^d$ and let $\phi : \mathbb{R} \to [0,1]$ with $\phi(1) \geq 1$. For any $v \in \mathbb{N}$ there exists a class hypothesis class H over $\mathcal{X}$ with VC-dimension v such that*

$$m[\mathcal{Q}^d_{\phi,\mathrm{det}}, H, \mathrm{proper}] = \Omega(1/\epsilon^2).$$

Proof. Consider two points x and y at distance 1 and let H be such that for every $h \in H$, $h(x) = h(y)$. A distribution in $\mathcal{Q}^d_{\phi,\mathrm{det}}$ can give two different labels to these points. Then, estimating a bias of $1/2 \pm \epsilon$ on the weight of these two points requires a sample size of $\Omega(1/\epsilon^2)$. See Section 4.4.2 in Chapter 4 for a more precise argument of this type. □

3.5 Discussion

In this chapter, we formally defined Probabilistic Lipschitzness, a data assumption that is relevant for several results in this thesis. We view PL as a property that formally captures the cluster assumption, a notion that is often invoked to motivate practical methods for learning with unlabeled data. In contrast to other data-assumptions or notions of prior knowledge that have been employed in theoretical studies in this area, PL is a property of the data generating distribution only. It does not involve any notion of compliance with a specific hypothesis class and is used in the analysis of our algorithms only.

Machine learning is often preceded by a process of feature selection or feature construction. Probabilistic Lipschitzness complies with an intuition of how such features should behave and can thus also provide a measure for the quality of a feature representation. It would be intriguing to investigate if PL, or a similar notion, could serve as an objective for developing a theoretical framework for feature learning.

Section 3.3.3 suggests a way to generalize the Probabilistic Lipschitzness condition. Instead of just measuring the mass of (even only mildly) heterogeneous neighborhoods, we could consider the function that assigns every point the heterogeneity of a neighborhood around it. Interpreting this function as a new labeling function and imposing a low noise condition (like the Tsybakov noise condition) provides a measure that is weaker than PL, but captures a similar property. It would be interesting to see if the results, that are here derived under the assumption of PL, generalize to this case.

Chapter 4

Proper Semi-Supervised Learning

4.1 Overview

Semi-Supervised Learning (SSL) is a framework of learning a classifier from a small number of labeled examples and many unlabeled examples. The goal of SSL is to use the available unlabeled data to save label supervision. This is widely used in practice. However, from a theoretical point of view, the question whether (or under which conditions) unlabeled data can *provably* be beneficial for reducing the number of labels needed for learning, is not well understood yet. (We discuss previous work on this topic in the next section.)

Most work on SSL focuses on how using unlabeled data can improve the accuracy of the learned classifiers. Here, we analyze how unlabeled data can be beneficiary for constructing better classifiers in other respects. We propose an algorithmic framework for semi-supervised learning that utilizes unlabeled examples for learning classifiers that satisfy specific, user-defined, requirements. For example, an application might require the learned classifier to run fast at prediction time. Or, a user might be interested to learn a label predictor that is easily interpretable. Linear separators (halfspaces) are examples of classifiers that satisfy both these requirements. Therefore, we particularly explain how our framework applies to learning halfspaces. However, the results of the chapter apply to (and are derived for) learning arbitrary classes of finite VC-dimension.

We formally analyze semi-supervised learning under such requirements. We propose an algorithmic framework for this setting and present conditions under which we provably save labels in comparison to standard (fully supervised) learning. Most of the content of this chapter appeared in a publication by Urner et al. [2011].

Outline of our results We model learning with requirements on the output classifier as proper learning (see Definition 6). That is, we specify a collection of predictors H, consisting of the classifiers that meet the prior requirements. We call this collection H the *output class*. Our goal is to find a low error classifier that is a member of the output class H. In many cases such a restriction renders the learning task harder. If the learner has some prior knowledge about the learning task, he can exploit this prior knowledge to learn some classifier with only a small number of samples. However, this classifier may not necessarily comply with the requirements, that is it may not be from the class H. Thus, if the learner is required to output a classifier from H, he might require more examples despite the prior knowledge about the task.

We propose to first ignore this restriction to the output class and use the labeled sample to solve the unrestricted learning task, that is to make full use available prior knowledge and learn some classifier of low error. As a second step, we use the unlabeled data to transform the learned classifier into a predictor from H. More concretely, our learning algorithms follow a, rather simple, 2-step paradigm:

1. Use the labeled sample to learn a classifier that is not necessarily a member of the output class H, but has small prediction error.

2. Apply that learned classifier to label the points of the unlabeled input sample, and feed that now labeled sample to a fully supervised H-learner.

Since labeled samples are needed only for the first step of this paradigm, the labeled sample complexity of our algorithm equals the sample complexity of the unrestricted learning task (in the standard framework). Consequently, whenever the search for an unrestricted low error classifier has a relatively low sample complexity, our SSL paradigm allows us to find a low error predictor *in the output class H* with fewer labeled-examples than what is required by any fully supervised algorithm for properly learning H.

We investigate two scenarios where such a saving of labeled examples occurs. In Section 4.4, we analyze our algorithmic paradigm when, on top of the output class H, which is determined by the task, the learner is aware of a different (possibly larger) class, *the accurate class H'*, that contains a low error classifier. Then, in Section 4.5, we investigate an alternative approach, that does not require the prior knowledge of such an H'. Instead, it uses the Nearest Neighbor algorithm in the first step of our paradigm. This second version is a sensible approach if the learner has good reasons to assume that the data is clusterable.

We now start with discussing relevant previous work (Section 4.2). We introduce a formal framework for SSL in Section 4.3 and then proceed to present the two scenarios mentioned above.

4.2 Related Work

Semi-supervised learning is a very active research area. Probably the most prolific direction is the introduction of algorithmic approaches and describing their application to real life learning tasks. Both Chapelle et al. [2006] and Zhu [2008] provide an overview on various algorithmic techniques that are used for semi-supervised learning in practice. Most of the work along these lines emphasizes experimental results and is not supported by formal performance guarantees.

Among practically oriented studies most relevant to our work on proper SSL, is a study by Liang et al. [2008]. This work also deals with the setup of using unlabeled data to improve the runtime of the learned classifier. It discusses learning for Natural Language Processing (NLP) tasks, where expressive conditional random field (CRF) predictors have low error but are slow to compute. It proposes to use unlabeled data to replace the CRF predictors by fast computable Independent Logistic Regression (ILR) classifiers. Liang et al. [2008] provide an upper bound on the sample complexity of the procedure, but do not address the reduction of sample sizes that is due to the use of unlabeled samples.

Their algorithmic paradigm is similar to ours and, in a sense, our work here can be viewed as generalizing that work. Prior to that, Bucila et al. [2006] showed experimentally how a similar idea can be applied to replace complex ensembles by compact neural network predictors.

At the other end of the spectrum there is purely theoretical research that focuses on abstract models. There are two lines along which SSL has been theoretically analyzed. In the first setting there are specific assumptions on the data generating distribution. Castelli and Cover compare the value of labeled and unlabeled samples in specific generative models, where they assume that the learner knows the distributions generating each label-class, but not the mixing-parameter (Castelli and Cover [1995], Castelli and Cover [1996]).

The other direction of theoretical research on SSL deals with methods that do not impose any prior assumptions about the data-generating distributions. Kääriäinen [2005] shows that in the realizable case, access to unlabeled data can reduce the error of the learned classifier by a factor of 2. In this setting, the unlabeled data is used to choose a classifier, among the empirical risk minimizers, that minimizes the maximum distance to another classifier in this set. However, it remains open in which cases such a classifier actually exists so that the error bound would be reduced. Furthermore, the work of Ben-David et al. [2008], shows that, even for the simple case of learning initial segments on the unit interval, unlabeled data cannot reduce the error by more than a factor of 2, if the marginal of the underlying distribution is a continuous function.

Balcan and Blum [2005] propose a framework for SSL that allows for PAC learning analysis and sample complexity bounds. They suggest to use a notion of a compatibility function that assigns a higher score to classifiers which "fit nicely" with respect to the marginal distribution. If compatible classifiers can be identified by unlabeled samples, these can be used to reduce the hypothesis space to only highly compatible classifiers, which in turn might reduce the labeled sample complexity for learning this now smaller class. However, it remains open whether (or under which conditions) a significant decrease of the hypothesis space actually occurs. Furthermore, their results also concern the realizable case only.

The negative results of Ben-David et al. [2008] imply that unlabeled data can only be provably beneficial under assumptions on the relationship between the marginal distribution and the labels. A popular assumption in that context is the so-called cluster assumption, postulating that the data contains clusters that have homogeneous labels. Under such an assumption, SSL works by using the unlabeled observations to identify these clusters, and then considering only classifiers that are constant on each cluster. Closely related to the cluster assumption are the *smoothness assumption* and the related *low density assumption* which suggests that the classification decision boundaries should lie in low density regions (Chapelle and Zien [2005]).

There is a line of work that proposes practical algorithms inspired by this intuition and provides some analysis of the proposed algorithms under this assumption. For example, Rigollet [2007] provides a mathematical formalization of (a version of) the cluster assumption. This work assumes that the data contains a collection of countably many connected components of high density (with respect to some threshold density level) and that the Bayes optimal classifier of the distribution is constant on those clusters. This work then restricts its attention to learning classifiers that also label the set of identified clusters homogeneously and analyses the benefits of unlabeled data for this task. The risk of reducing the set of potential classifiers according to the clusterability assumption in this manner, is that the learner may be left with only poorly performing classifiers, if the presumed label-structure relationship fails to hold.

4.3 Formal Framework for SSL

In this section we provide a formal framework for proper SSL learning.

As in standard learning (see Chapter 2), a learning task is modeled as a distribution P over $\mathcal{X} \times \{0,1\}$, where $\mathcal{X}$ is some domain set. In this chapter, we assume that the labeling function l of the distribution P is deterministic, (i.e. $l(x) \in \{0,1\}$ for all $x \in \mathcal{X}$). Given a hypothesis class $H \subseteq \{0,1\}^{\mathcal{X}}$, an *H-proper SSL learner* $\mathcal{A}$ takes a labeled sample S, generated *i.i.d.* by P, and an unlabeled sample T, generated *i.i.d.* by $P_{\mathcal{X}}$, and outputs a function $h \in H$. Formally,

$$\mathcal{A} \ : \ \left(\ \bigcup_{m=1}^{\infty} (\mathcal{X} \times \{0,1\})^m \ \times \ \bigcup_{n=1}^{\infty} \mathcal{X}^n \ \right) \ \to \ H \ .$$

We now present notions of learnability and label-complexity for proper semi-supervised learning. These are adaptations of our Definitions 6 and 2 to this setting.

Definition 24 (Proper SSL Learning)**.** We say that an SSL learner $\mathcal{A}$ *properly learns* some hypothesis class H over $\mathcal{X}$ with respect to a set of distribution $\mathcal{Q}$ over $\mathcal{X} \times \{0,1\}$, if there exist functions $m : (0,1) \times (0,1) \to \mathbb{N}$ and $n : (0,1) \times (0,1) \to \mathbb{N}$ such that, for all distributions $P \in \mathcal{Q}$, for all $\epsilon > 0$ and $\delta > 0$, when given an *i.i.d.* (labeled) sample of size at least $m(\epsilon, \delta)$ from P and an *i.i.d.* (unlabeled) sample of size at least $n(\epsilon, \delta)$ from $P_{\mathcal{X}}$, then, with probability at least $1 - \delta$ over the samples, $\mathcal{A}$ outputs a classifier $h \in H$ with error at most $\mathrm{Err}_P(H) + \epsilon$. In this case, we also say that the SSL learner *properly* (ϵ, δ)*-learns* H with respect to $\mathcal{Q}$.

It is commonly assumed that unlabeled data is abundantly available, and therefore we are not as concerned with the size of the unlabeled sample in semi-supervised learning as we are with bounding the number of labeled examples we use. The following definition of the *labeled sample complexity* of proper SSL learning reflects this emphasis on the size of the labeled sample. We define the labeled sample complexity of an algorithm (or a learning task) as the smallest function m for the size of the labeled sample for which there exists some (possibly large) function n for the size of the unlabeled sample such that m and n satisfy the condition in Definition 24. Alternatively, one could define the labeled sample complexity stagewise, first with respect to a fixed function n, and then taking the minimum over all these. To avoid unneeded notation, we chose to omit this step. However, we do provide concrete bounds on the unlabeled sample sizes in all of our results (as opposed to mere existence proofs that would also comply with the definition below).

Definition 25 (Labeled Sample Complexity of Proper SSL Learning)**.** We call the smallest function $m : (0,1) \times (0,1) \to \mathbb{N}$ for which there exist a function $n : (0,1) \times (0,1) \to \mathbb{N}$ such that m and n together satisfy the condition of Definition 24 the *labeled sample complexity of the SSL learner $\mathcal{A}$ for properly learning H with unlabeled data with respect to $\mathcal{Q}$*. We denote this function by $m^{\mathrm{ssl}}[\mathcal{A}, \mathcal{Q}, H, \mathrm{proper}]$. We call the smallest function $m : (0,1) \times (0,1) \to \mathbb{N}$ such that there exists an SSL learner $\mathcal{A}$ with $m^{\mathrm{ssl}}[\mathcal{A}, \mathcal{Q}, H, \mathrm{proper}] \leq m$ the *labeled sample complexity of properly learning H with unlabeled data with respect to $\mathcal{Q}$* and denote this function by $m^{\mathrm{ssl}}[\mathcal{Q}, H, \mathrm{proper}]$.

4.4 Proper SSL Using a Large Accurate Class

One scenario in which we can make use of unlabeled data for learning a classifier from H, is the learner having prior knowledge of a richer class of predictors H' that has low (or even zero) approximation error with respect to the data-generating distribution. To allow for label savings, its approximation error needs to be much lower than that of the best predictor in the class H. We call this class H' the *accurate class*.

If the data generating distribution is realizable (see Definition 4) by a class H' of finite VC-dimension, we know that it can be learned (in the standard, fully supervised, setup) with a sample complexity of $C_2 \frac{\mathrm{VCdim}(H') \ln(1/\epsilon) + \ln(1/\delta)}{\epsilon}$ with the constant C_2 from Theorem 16. In this case, our algorithm first learns a low error classifier from this accurate class H' (with the stated number of labeled examples), and then uses the learned classifier to label an unlabeled sample and learn a classifier from the output class H with this. The labeled sample complexity of our paradigm is $O(\frac{1}{\epsilon})$ for any such pair of accurate class H' and output class H.

On the other hand, we prove a lower bound of $\Omega(\frac{1}{\epsilon^2})$ labeled samples, for certain pairs of output class H and accurate class H'. Note that we cannot simply invoke the lower bound for proper learning from Corollary 17. For the upper bound on the labeled sample complexity that we present here, the learner has the additional prior knowledge of realizability by the accurate class H'. Therefore, a fair comparison requires establishing a lower bound on the labeled sample complexity under those same conditions. We prove that a lower bound of $\Omega(\frac{1}{\epsilon^2})$ holds for the task of learning linear separators (as the output class) without unlabeled examples for distributions that are realizable by some learnable accurate classes H'. It follows that in such scenarios, the labeled sample complexity of our SSL learning algorithm is strictly lower than that of any fully supervised proper learner for the output class H of linear classifiers.

4.4.1 Upper bound

Let H be the output class of our H-proper SSL learner and let H' be an accurate class. We denote by $\mathcal{A}_{(H,H')}$ the version of our two stage SSL algorithm, that first learns H' using the labeled examples and then learns H using the predictions of the previous stage on unlabeled examples.

We first derive an upper bound on the labeled sample complexity of $\mathcal{A}_{(H,H')}$. For this, we need the following lemma that states the robustness of agnostic learners (see Definition 5) to small shifts in the data generating distribution. More concretely, it states that if an agnostic learner gets a sample that was labeled by a low-error function, instead of the true labeling function of the distribution, the error of its output will not deteriorate too much. This will allow us to use the function that is learned in the first step of our paradigm for labeling some unlabeled sample and feed this to an agnostic learner.

Recall that $m[\mathcal{A}, H](\epsilon, \delta)$ denotes the sample size required by $\mathcal{A}$ to (ϵ, δ)-learn H for an agnostic learner $\mathcal{A}$.

Lemma 26. *Let P be a distribution over $\mathcal{X} \times \{0, 1\}$, let $f : \mathcal{X} \to \{0, 1\}$ be a function with $\mathrm{Err}_P(f) \leq \epsilon$, let $\mathcal{A}$ be an agnostic learner for some hypothesis class H over $\mathcal{X}$. Then, if given an i.i.d. sample of size $m[\mathcal{A}, H](\epsilon, \delta)$ from P's marginal $P_{\mathcal{X}}$ labeled by f, $\mathcal{A}$ outputs a hypothesis h with*

$$\mathrm{Err}_P(h) \leq \mathrm{opt}_H(P) + 3\epsilon$$

with probability at least $(1 - \delta)$.

Proof. Let P' be the distribution that has the same marginal as P (i.e. ${P'}_{\mathcal{X}} = P_{\mathcal{X}}$) and f as its deterministic labeling rule. We first show that

$$|\mathrm{Err}_P(h) - \mathrm{Err}_{P'}(h)| \ \leq \ \epsilon \tag{4.1}$$

holds for any classifier $h : \mathcal{X} \to \{0, 1\}$. We have

$$\begin{aligned}
\mathrm{Err}_P(h) &= \Pr_{(x,y)\sim P}[y \neq h(x)] \\
&\leq \Pr_{(x,y)\sim P}[y \neq h(x) \wedge y = f(x)] + \Pr_{(x,y)\sim P}[y \neq h(x) \wedge y \neq f(x)] \\
&\leq \mathrm{Err}_{P'}(h) + \epsilon,
\end{aligned}$$

and similarly

$$\begin{aligned}\mathrm{Err}_{P'}(h) &= \Pr_{x\sim P_{\mathcal{X}}}[f(x)\neq h(x)]\\ &= \Pr_{(x,y)\sim P}[f(x)\neq h(x)]\\ &\leq \Pr_{(x,y)\sim P}[f(x)\neq h(x)\wedge y=f(x)] + \Pr_{(x,y)\sim P}[f(x)\neq h(x)\wedge y\neq f(x)]\\ &\leq \mathrm{Err}_P(h)+\epsilon.\end{aligned}$$

This implies Equation 4.1. Now, for the optimal hypothesis h^* in H with respect to P we get

$$\mathrm{Err}_{P'}(h^*) \;\leq\; \mathrm{Err}_P(h^*)+\epsilon \;=\; \mathrm{opt}_P(H)+\epsilon,$$

and thus

$$\mathrm{opt}_{P'}(H) \;\leq\; \mathrm{Err}_{P'}(h^*) \;\leq\; \mathrm{opt}_P(H)+\epsilon.$$

This implies, that when we feed an *i.i.d.* sample S of size $m[\mathcal{A},H](\epsilon,\delta)$ from P' to the agnostic learner $\mathcal{A}$, it outputs a classifier h with

$$\mathrm{Err}_{P'}(h) \;\leq\; \mathrm{opt}_{P'}(H)+\epsilon \;\leq\; \mathrm{opt}_P(H)+2\epsilon$$

with probability at least $(1-\delta)$, where the first inequality follows from the definition of an agnostic learner. The last inequality implies for this function h that

$$\mathrm{Err}_P(h) \;\leq\; \mathrm{Err}_{P'}(h)+\epsilon \;\leq\; \mathrm{opt}_H(P)+3\epsilon.$$

□

Using this lemma, it suffices to prove that the output of the first step of the algorithm on the labeled sample S results in a classifier of small probability of error. Then, when we feed the unlabeled sample T labeled by this function to an agnostic proper learner $\mathcal{A}$ for H, this learner is still guaranteed to output a good predictor. More precisely, for each of our positive results, we will argue that the first step outputs a classifier of error at most $\epsilon/3$ with probability at least $1-\delta/2$. Then, if we label an unlabeled sample of size $m[\mathcal{A},H,\mathrm{proper}](\epsilon/3,\delta/2)$ with this classifier and use this to agnostically learn H with

$\mathcal{A}$ in the second step, Lemma 26 implies that our paradigm outputs a classifier from H of error at most ϵ with probability at east $1-\delta$.

With this, we obtain the following result for the labeled sample complexity of $\mathcal{A}_{H,H'}$:

Theorem 27. *There exists a constant $C > 0$ such hat for every pair of hypothesis classes H, H', every $\epsilon, \delta \in (0,1)$ and every distribution P over some domain $\mathcal{X} \times \{0,1\}$, if $\mathrm{opt}_{H'}(P) = 0$ then, given access to a labeled sample S of size*

$$C\frac{\mathrm{VCdim}(H')\ln(1/\epsilon) + \ln(1/\delta)}{\epsilon},$$

and an unlabeled sample T of size

$$C\frac{\mathrm{VCdim}(H) + \ln(1/\delta)}{\epsilon^2},$$

with probability at least $(1-\delta)$, we have

$$\mathrm{Err}_P(\mathcal{A}_{(H,H')}(S,T)) \leq \mathrm{opt}_H(P) + \epsilon.$$

Proof. To prove the bound, we make use of standard results of VC-theory. We set C' to be the constant C_2 from Theorem 16. Then, as H' is realizable, Theorem 16 tells us that a sample size of

$$C'\frac{\mathrm{VCdim}(H')\ln(3/\epsilon) + \ln(2/\delta)}{\epsilon/3}$$

suffices for ERM(H') to output a classifier from H' that has error at most $\epsilon/3$ with probability at least $1-\delta/2$. Similarly, for ERM(H) a labeled sample of size

$$C'\frac{\mathrm{VCdim}(H) + \ln(2/\delta)}{(\epsilon/3)^2}$$

suffices to output a classifier with error at most $\epsilon/3$ with probability at least $1-\delta/2$. Now Lemma 26, and setting C to substitute ϵ for $\epsilon/3$ in the above formulas, implies the claim. □

Note that, even in cases where $\mathrm{VCdim}(H) < \mathrm{VCdim}(H')$, for small enough values of ϵ this upper bound on the sufficient size of the labeled sample is smaller than the known lower bound of $\Omega\left(\frac{\mathrm{VCdim}(H)+\log(1/\delta)}{\epsilon^2}\right)$ on the required labeled sample size for learning a hypothesis from H in the agnostic setup when no additional unlabeled sample is available. We show in the next subsection, that there are specific distributions for which we can prove the lower bound of $\Omega(\frac{1}{\epsilon^2})$ on the sample complexity of learning from labeled examples only, even under the additional assumption of realizability by some accurate class H'.

4.4.2 Lower bound

For our lower bound, we let H be the class of halfspaces in $\mathbb{R}^d$ and let H' be the class of all unions of members of H. A halfspace $h_w : \mathbb{R}^d \to \{0,1\}$ is a classifier that labels each $x \in \mathbb{R}^d$ by 1 if $\mathrm{sign}(wx)$ is positive and by 0 otherwise, where $w \in \mathbb{R}^d$ is a vector defining the halfspace and wx denotes the inner product between w and x. A union of two halfspaces h_w and h_v is a classifier h_{wv} that labels an element $x \in \mathbb{R}^d$ with 1 if and only if at least one of h_w and h_v assigns label 1 to x.

Further, we let $\mathcal{Q}$ be the family of all data probability distributions over $\mathbb{R}^d \times \{0,1\}$ that are realizable by H'.

Theorem 28. *The sample complexity of (ϵ,δ)-agnostically learning halfspaces in $\mathbb{R}^d$ with respect to the set of distributions $\mathcal{Q}$, is bounded from below by*

$$\frac{1-(4\epsilon)^2}{2(4\epsilon)^2}\ln\left(\frac{1}{8\delta(1-2\delta)}\right).$$

The rest of this subsection is devoted to presenting the proof of this theorem. Our proof follows the idea of the proofs of lower bounds for the sample complexity of learning in Chapter 5 of Anthony and Bartlett [1999].

Let $\mathcal{P}_T$ be the set of probability distributions that have the discrete set

$$T = \{(z_1,1),(z_2,0),(z_3,1)\}$$

of three distinct, collinear points as their support. We require z_1, z_2 and z_3 to lie on a line in $\mathbb{R}^d$ and z_2 to lie between z_1 and z_3 on this line. Note that $\mathcal{P}_T \subseteq \mathcal{Q}$, as the labeling on z_1, z_2 and z_3 can be realized by a union of two halfspaces.

We now describe the main idea of our proof: Clearly, no halfspace can label all of these three point correctly, so the choice of the optimal classifier in H depends on the relative weights that the input distribution assigns to these points. Having access to a large sample (labeled or unlabeled) will allow approximating these weights to arbitrary accuracy (as the size of the sample goes to infinity), and therefore a labeled sample that contains all three of these points will allow detecting the optimal H predictor. In case the weights of z_1 and z_3 are roughly the same, say the difference is only ϵ, we can adapt the lower bound from Lemma 5.1 in Anthony and Bartlett [1999] (fully cited as Theorem 29 below) to show that we need at least $\frac{1-(1.5\epsilon)^2}{2(1.5\epsilon)^2} \ln\left(\frac{1}{8\delta(1-2\delta)}\right) = \Omega(1/\epsilon^2)$ sample points to decide which point has more weight. Thereby we show that, without access to any unlabeled samples, the number of labeled sample points required to properly learn H with error below $\text{opt}_P(H) + \epsilon$ is $\Omega(1/\epsilon^2)$

To formally prove our lower bound, we consider the following problem and reduce it to our learning problem:

The two-coin-problem with parameter ϵ: There are two coins, for one the probability of Heads is $1/2 + \epsilon/2$ and for the other it is $1/2 - \epsilon/2$. Let us denote the coin with higher heads probability C_H and the other coin C_T. One of these coins is tossed m times. A learner gets the outcome of these m coin tosses, and has to determine which of the two coins was used.

Theorem 29 (Lemma 5.1 in Anthony and Bartlett [1999])**.** *If the coin is picked randomly, with probability* 0.5 *for* C_H *and* 0.5 *for* C_T*, then, for any* $0 < \delta < 1/4$ *and any* $0 < \epsilon < 1$*, if the number of seen coin tosses,* m*, is less than*

$$2\frac{1-\epsilon^2}{2\epsilon^2} \ln\left(\frac{1}{8\delta(1-2\delta)}\right)$$

then any algorithm for the two-coin problem (i.e., a function that takes the outcome of the coin tosses as input and outputs either C_H *or* C_T*) has probability of error larger than* δ*.*

Reduction of the two-coin problem to properly learning H with respect to $\mathcal{P}$:

Given a series $(c_1, \dots, c_m)$ of outcomes of coin tosses, $c_i \in \{\text{Heads, Tails}\}$, we generate a labeled sample $S = ((x_1, y_1), \dots (x_m, y_m))$ for our learning problem as follows:

1. Define a function $f : \{\text{Heads, Tails}\} \to T$ by $f(\text{Heads}) = (z_1, 1)$ and $f(\text{Tails}) = (z_3, 1)$.

2. Draw a an i.i.d. sample $p_1, \ldots, p_m$ of size m from a Bernoulli distribution, where each p_i is 1 with probability 2/3 and 0 with probability 1/3.

3. To construct the sample S, set (x_i, y_i) to be $f(c_i)$ if $p_i = 1$, and set (x_i, y_i) to be $(z_2, 0)$ whenever $p_i = 0$.

4. Finally, let $S^*(c_1, \ldots, c_m)$ be $((x_1, y_1), \ldots, (x_m, y_m))$.

The operator S^* defined above transforms a series of coin tosses to a sample from the set T. Depending on the coin that generated the series of coin tosses, this sample can be viewed as an *i.i.d.* sample from one of the following two distributions:

- Either from $S^*(C_H)$, the distribution over T with $\Pr[(z_1, 1)] = (1+\epsilon)/3$, $\Pr[(z_3, 1)] = (1-\epsilon)/3$ and $\Pr[(z_2, 0)] = 1/3$,

- or from $S^*(C_T)$, the distribution over T with $\Pr[(z_1, 1)] = (1-\epsilon)/3$, $\Pr[(z_3, 1)] = (1+\epsilon)/3$ and $\Pr[(z_2, 0)] = 1/3$.

Note that both these distributions are members of our class of distributions $\mathcal{P}_T$.

Lemma 30. *For any $\epsilon, \delta \in (0, 1)$ and any $m \in \mathbb{N}$ the following holds: If there exists an algorithm $\mathcal{A}$ that $(\epsilon/4, \delta)$-properly learns H over T from m labeled examples (without unlabeled samples), then there also exists an algorithm $\mathcal{A}_C$ that solves the two-coin problem with parameter ϵ correctly from m coin tosses with probability greater than $(1 - \delta)$.*

Proof. Given the result of m *i.i.d.* coin tosses, $(c_1, \ldots, c_m)$, the labeled sample $S^*((c_1, \ldots, c_m))$ is an *i.i.d.* sample from the distribution $S^*(C_H)$ or from the distribution $S^*(C_T)$, depending on which coin was chosen. Now, given a learning algorithm, $\mathcal{A}$, for distributions supported on T that outputs hypotheses from H, we define an algorithm $\mathcal{A}_C$ for the two-coin problem by setting $\mathcal{A}_C(c_1, \ldots, c_m) = C_H$ if and only if $\mathcal{A}(S^*((c_1, \ldots, c_m))(z_1) = 1$; that is, if the halfspace that $\mathcal{A}$ outputs on the input $S^*((c_1, \ldots, c_m))$ labels the point z_1 positively.

It is easy to see that, for each of the distributions $S^*(C_H)$ and $S^*(C_T)$, the true error of the optimal predictor from H is $(1-\epsilon)/3$. However, whenever $\mathcal{A}_C$ predicts the wrong coin, the error of the hypothesis $\mathcal{A}(S^*((c_1,\dots,c_m)))$ is at least $1/3$ (it is $1/3$ if the halfspace $\mathcal{A}(S^*((c_1,\dots,c_m)))$ labels both z_1 and z_3, and thus also z_2, with 1; it is $(1+\epsilon)/3$ if it labels the lighter point with 1 but z_2 with 0; and it is larger in all other cases). Now note that $1/3 = \mathrm{Err}_P(H) + \epsilon/3 > \epsilon/4$, where $P \in \{S^*(C_H), S^*(C_T)\}$ is the distribution induced by the coin that was tossed. □

Now Lemma 30 and Theorem 29 imply Theorem 28.

4.4.3 Reduction of label complexity

The scenario analyzed above demonstrates that there are situations in which access to unlabeled samples provably reduces the labeled sample complexity of proper learning. Consider the set $\mathcal{P}_T$ of distributions over the set T from the previous section. For a concrete example, let H' be the class of unions of halfspaces and H be the class of halfspaces over $\mathbb{R}^2$. The class H' has VC-dimension 5 and every distribution over T is realizable by H'. Therefore the upper bound from Theorem 27 applies. Thus, using $\mathcal{A}_{H,H'}$, we can properly learn H, which has VC-dimension 3, with a labeled sample of size $C\frac{5\ln(1/\epsilon)+\ln(1/\delta)}{\epsilon}$, provided that we have also access to an unlabeled sample of size $C\frac{3+\log(1/\delta)}{\epsilon^2}$, where C is the constant from Theorem 27.

Combining this with the lower bound of $\Omega(1/\epsilon^2)$ in Theorem 28 for properly learning H under the same conditions from labeled samples only, yields that in this situation the use of unlabeled data is provably beneficial, as our algorithmic paradigm requires less labeled data than any fully supervised proper learner.

Corollary 31. *Let et H' be some class of finite VC-dimension over $\mathbb{R}^d$ such that the class $\mathcal{Q}_{H'}$ of distributions that are realized by H' contains the set $\mathcal{P}_T$ of distributions from Section 4.4.2 for some set of collinear points $T = \{z_1, z_2, z_3\}$. Consider the task of proper learning the class H of halfspaces in $\mathbb{R}^d$ with respect to $\mathcal{Q}_{H'}$. For this task, the labeled sample complexity of our SSL algorithm $A_{(H,H')}$ is strictly below the labeled sample complexity of* any *proper learning algorithm in the standard framework (that does not have access to unlabeled examples).*

4.5 Proper SSL Using Nearest Neighbors

We now investigate our paradigm when the first step is carried out using a Nearest Neighbor algorithm. The obvious advantage of this version over the SSL algorithm discussed above is that it does not require the prior knowledge of any class of low (or even zero) approximation error. The Nearest Neighbor algorithm (NN) takes a labeled sample and, when required to label some domain point, assigns to it the label of its nearest neighbor in the labeled sample (with respect to some underlying metric). See Section 2.3.1 for a precise definition.

We investigate the sample complexity of the NN-based SSL paradigm as a function of the degree by which the learning task conforms with the cluster assumption, as quantified by our probabilistic Lipschitzness measure. We first consider the case that the data distribution can be partitioned into homogeneously labeled clusters with some margin separating any pair of differently labeled clusters. Recall that, if this is the case, the labeling function satisfies the standard Lipschitz condition (Definition 8). We show that in this case the labeled sample complexity of our algorithm is $O(1/\epsilon)$ (for any learnable output class H, and, in particular, for the task of prediction with linear halfspaces).

We then proceed to show that the NN-based SSL algorithm can also be beneficial under milder conditions. In particular, if the underlying data distribution satisfies some probabilistic Lipschitzness condition, the labeled sample complexity of our SSL algorithm is $C \cdot \frac{2}{\epsilon\delta}\left(\frac{\sqrt{d}}{\phi^{-1}(\epsilon)}\right)^d$ for some constant C, where d is the dimension of the Euclidean space in which our data is embedded, for any learnable output class H.

Finally, we again argue that properly learning the class of linear separators without access to unlabeled data requires a (labeled) sample size of $\Omega(\frac{1}{\epsilon^2})$ even under certain assumptions on the probabilistic Lipschitzness. We then combine the upper and lower bounds on the labeled sample complexity of proper learning under these conditions to establish the benefits of unlabeled data for proper learning under probabilistic Lipschitzness for specific PL-functions ϕ.

4.5.1 Upper bounds

In this section we provide upper bounds on the labeled sample complexity of our algorithmic paradigm when a Nearest Neighbor algorithm is used in the first step. We denote this version of our paradigm by $A_{NN}(S, T)$.

We begin by considering the scenario where the data is scattered in separated, homogeneously labeled clusters. It is easy to see that under this assumption, the Nearest Neighbor algorithm will have close-to-zero error if the input sample is large enough to hit (almost) each of these clusters. As explained in Section 2.3.1, this situation can be modeled by requiring that the labeling function is L-Lipschitz (where $1/L$ is a lower bound on the separation between any two differently labeled clusters). Recall that $\mathcal{Q}^d_{L,\mathrm{det}}$ denotes the set of distributions over $[0,1]^d \times \{0,1\}$ with a deterministic, L-Lipschitz labeling function.

Theorem 32. *Let $\mathcal{X} = [0,1]^d$ be the unit cube of $\mathbb{R}^d$ and let H be a hypothesis class of finite VC-dimension over $\mathcal{X}$. Then, there exists a constant C such that, for all $\epsilon, \delta \in (0,1)$, for all $L > 0$, and for every distribution $P \in \mathcal{Q}^d_{L,\mathrm{det}}$, we have with probability at least $1-\delta$ over an i.i.d. labeled sample S of size at least*

$$\left(L\sqrt{d}\right)^d \frac{6}{\epsilon\delta \mathrm{e}}$$

and an i.i.d. unlabeled sample T of size at least

$$C\frac{\mathrm{VCdim}(H) + \ln(1/\delta)}{\epsilon^2}$$

that

$$\mathrm{Err}_P(A_{NN}(S,T)) \leq \mathrm{Err}_P(H) + \epsilon.$$

Proof. This result follows from Lemma 30 by noting the following:

1. A sample size of $\left(L\sqrt{d}\right)^d \frac{6}{\epsilon\delta \mathrm{e}}$ suffices to output classifier of error at most $\epsilon/3$ with confidence at least $1-\delta/2$ (see Theorem 13).
2. $C\frac{\mathrm{VCdim}(H)+\ln(1/\delta)}{\epsilon^2} = m[H, \mathrm{proper}](\epsilon/3, \delta/2)$ is the sample complexity of properly $(\epsilon/3, \delta/2)$-learning H for a suitable constant C (see Corollary 17).

□

Thus, if the data lies in well separated clusters, the required labeled sample size grows only linearly with $1/\epsilon$. Next, relaxing this condition, we assume that the probability of two differently labeled points decays smoothly as they get closer. Recall that $\mathcal{Q}^d_{\phi,\mathrm{det}}$ denotes the class of all distribution over $[0,1]^d \times \{0,1\}$ whose labeling function is deterministic and satisfies the ϕ-probabilistic Lipschitzness.

Theorem 33. *Let $\mathcal{X} = [0,1]^d$ be the unit cube of $\mathbb{R}^d$, let H be a hypothesis class of finite VC-dimension over $\mathcal{X}$ and let $\phi : \mathbb{R} \to [0,1]$ be some PL-function. Then, there exists a constant C such that, for all $\epsilon, \delta \in (0,1)$, and for every distribution $P \in \mathcal{Q}^d_{\phi,\mathrm{det}}$, we have with probability at least $1-\delta$ over an i.i.d. labeled sample S of size at least*

$$\frac{2}{\epsilon\,\delta\,\mathrm{e}}\left(\frac{\sqrt{d}}{\phi^{-1}(\epsilon)}\right)^d$$

and an unlabeled i.i.d. sample T of size at least

$$C\frac{\mathrm{VCdim}(H) + \ln(1/\delta)}{\epsilon^2}$$

we have that

$$\mathrm{Err}_P(A_{NN}(S,T)) \leq \mathrm{opt}_H(P) + \epsilon.$$

Proof. The proof is identical to that of Theorem 32 except for here citing Theorem 20 for the sample complexity learning under probabilistic Lipschitzness for the size of the labeled sample. □

4.5.2 Lower bound

In this section we argue that we can adapt the lower bound for proper learning from Section 4.4.2 to the setting of proper learning with respect to distributions whose labeling function satisfies some probabilistic Lipschitzness. To adapt the construction from that section, we need to be able to place three collinear points z_1, z_2, z_3 in the space $[0,1]^d$ and set the labeling $l(z_1) = 1, l(z_2) = 0$ and $l(z_3) = 1$. In order for this labeling to comply with the probabilistic Lipschitz function ϕ (with the weights chosen in that construction), we need $\phi(|z_1 - z_3|/2) = 1$. If this holds, we can place z_2 at equal distance from z_1 and z_3 (between the two) and give it the required label. Note that the largest distance we can enforce between two points in $[0,1]^d$ is $\sqrt{d}$. Thus, we get the following result:

Corollary 34. *Let $\phi : \mathbb{R} \to [0,1]$ be a non-decreasing function with $\phi(\sqrt{d}/2) = 1$. Then the sample complexity of any standard proper learner for the class of halfspaces with respect to a set of distributions that includes $\mathcal{Q}^d_{\phi,\mathrm{det}}$ is at least $\Omega(\frac{1}{\epsilon^2})$.*

The above lower bound holds in particular for the function ϕ that corresponds to standard $(2/\sqrt{d})$-Lipschitzness. (Recall that this is the function ϕ with $\phi(\lambda) = 0$ of $0 \leq \lambda \leq \sqrt{d}/2$ and $\phi(\lambda) = 1$ for $\lambda > \sqrt{d}/2$.)

4.5.3 Reduction of labeled sample complexity

Theorem 32 provides an upper bound $O(1/\epsilon)$ on the number of labels that our algorithm requires to properly learn any class H of finite VC-dimension over $[0,1]^d$ with respect to distributions whose deterministic labeling function satisfies the standard Lipschitzess. Thus, the lower bound of Theorem 34 implies that, for the class of distributions with Lipschitz constant at most $L \leq 2/\sqrt{d}$, unlabeled data is provably beneficial for properly learning the class of halfspaces over $[0,1]^d$, as the labeled sample complexity gets reduced from $\Omega(1/\epsilon^2)$ to $O(1/\epsilon)$.

For the case of probabilistic Lipschitzness, we consider two cases for the function ϕ: First, we set $\phi(\lambda) = \lambda^n$ (polynomial Lipschitzness). In this case the upper bound on the labeled sample complexity in Theorem 33 becomes

$$\frac{2}{\epsilon\,\delta}\left(\frac{2^{1/n}\sqrt{d}}{\epsilon^{1/n}}\right)^d = O\left(\left(\frac{1}{\epsilon}\right)^{\frac{d+n}{n}}\right).$$

Thus, the labeled sample complexity of our paradigm is $o(1/\epsilon^2)$ if $n > d$. On the other hand, we have $\phi(\sqrt{d}/2) \geq 1$ for any n if $d \geq 4$. With this, the lower bound of Theorem 34 applies and we have established that unlabeled data provably reduces the labeled sample complexity of properly learning halfspaces over $[0,1]^d$, where $d \geq 4$, with respect to the class of distributions that satisfy the ϕ-probabilistic Lipschitzess for the function $\phi(\lambda) = \lambda^n$ with $n > d$.

Next, we consider $\phi(\lambda) = \mathrm{e}^{(\lambda-1)/\lambda} = \mathrm{e} \cdot \mathrm{e}^{-1/\lambda}$ (exponential Lipschitzness). In this case the upper bound on the labeled sample complexity in Theorem 33 becomes

$$\frac{2}{\epsilon\delta}\left(\sqrt{d}\ln(2\mathrm{e}/\epsilon)\right)^d \;=\; O\left(\frac{\ln(1/\epsilon)^d}{\epsilon}\right) \;=\; o\left(\frac{1}{\epsilon^2}\right).$$

Further, we have $\phi(\sqrt{d}/2) \geq 1$ if $d \geq 4$. Thus, the lower bound of Theorem 34 applies and we have established that unlabeled data provably reduces the labeled sample complexity of properly learning halfspaces over $[0,1]^d$, where $d \geq 4$, with respect to the class of distributions that satisfy the ϕ-probabilistic Lipschitzess for the function $\phi(\lambda) = \mathrm{e}^{(\lambda-1)/\lambda}$.

4.6 Discussion

We have presented an SSL framework for properly learning a hypothesis class of finite VC-dimension. Our paradigm is relevant, if the learner is confined to outputting a classifier of a certain type (for example due to interpretability or speed of computation requirements), but can more easily learn a low error classifier of a different type. We analyzed situations under which this framework provably saves labels in comparison to any fully supervised (standard) proper learner, in particular for the class of linear halfspaces.

While we have presented lower bounds only for learning linear halfspaces in this chapter, it is not difficult to see that one could construct similar lower bounds for other (pairs of) hypothesis classes. The lower bound construction uses that the larger (accurate) class H' can realize a labeling on some set that cannot be realized by the output class H, although H can shatter any subset (or realize any part of the labeling on a subset). In our case, the class of unions of halfspaces shatters a set of three collinear points while the class of halfspaces cannot realize any labeling on such points. Therefore, for determining the best halfspace, it becomes essential to estimate the weights of the points, so that a halfspace that misclassifies the lightest point can be identified as optimal. This construction can be applied to any other such situation resulting in the same reduction of labeled sample complexity for proper SSL learning. Such a more general construction has been presented by Ben-David and Ben-David [2011] in order to lower bound the sample complexity of proper learning in a setting where the learner knows the true labeling function in advance.

The upper bound on the size of the labeled sample in Theorem 27 corresponds to the sample complexity of learning the class H'. We apply it to learn a class H that has higher sample complexity due to having higher approximation error. However, it is easy to see that the same paradigm can be applied when the higher sample complexity of H is due to other reasons, such as having higher VC dimension. In fact, this situation occurs in the scenario discussed by Liang et al. [2008].

Both versions of our paradigm (the one using an accurate class and the one using Nearest Neighbors) have been experimentally evaluated by applying it to the MNIST data set (Urner et al. [2011]). The experiments confirm the theoretical results. They show that, in both cases, using unlabeled data reduces the number of labeled examples needed in order to learn linear classifier of a certain accuracy.

Chapter 5

Domain Adaptation

5.1 Overview

The standard model of machine learning focuses on a setting where the training and test data are generated by the *same* underlying distribution. While this may sometimes be a good approximation of reality, in many practical tasks this assumption cannot be justified. The data generating distribution might change over time or a learner may resort to using training data generated by a different distribution when labeled data available from the relevant target domain is hard to get by. The task of learning when the training and test data generating distributions differ is referred to as *Domain Adaption* (DA) learning. We call the distribution that generates the training data the *source distribution* and the distribution that the learned classifier is supposed to be applied to the *target distribution* (or *target task*).

Domain adaptation tasks occur in many practical situations and are frequently addressed in experimental research (see related work section for examples). For example, in natural language processing one might be able to access labeled documents of a certain type, say from legal documents, but needs to build a classifier to label the content of documents of a different type, say medical documents. In many such applications, while labeled data from the target distribution is not readily available, it is easy to get unlabeled examples from that target domain.

A domain adaptation setting can be characterized by the following aspects:

The input data available to the learner: A learner might get both labeled and unlabeled data from the source distribution (as well as, presumably in smaller numbers, from the target distribution). In line with most theoretical work on DA, we consider a setting where the learner has labeled data from a source distribution and unlabeled data from a target distribution available. Some DA learning approaches do not make use of the unlabeled target sample, and rather just learn a classifier from the source generated labeled sample, expecting that this classifier will also perform well on the target distribution. In this work, we call such DA learners *conservative*. A DA learner that actually employs the data available from the target distribution is called *non-conservative* or *adaptive*. We focus on analyzing the sample complexity of DA learning with respect to each of these types of samples (labeled source and unlabeled target sample). We present both conservative and adaptive DA paradigms.

The relationship between the source and target data generating distributions: It is not hard to see that domain adaptation learning is doomed to fail, if the source and target generation do not exhibit sufficient similarity. One task of a theoretical analysis of DA learning is thus to identify relatedness assumptions for source and target that allow DA learning to succeed. For our analysis, we assume the so-called *covariate shift* setting (Sugiyama and Mueller [2005]), that is, we assume that source and target distribution have the same labeling function. This assumption is often made in studies on the theory of DA and is justified for many practical applications (see Section 5.4.1 for a discussion). Besides the covariate shift assumption, we consider several measures for the relationship between the marginals of the source and target distribution. Mainly, we employ a bound on the ratio of target and source weights for certain collections of subsets of the domain. We show, that we can estimate this weight ratio from finite samples.

The prior knowledge about the task: Prior knowledge about a learning task is required for any guarantees of success (this is the No-Free-Lunch principle, see Theorem 7 and discussion). Since the goal of the learner is to come up with a low-error predictor for the target task, prior knowledge about that task is essential. For our positive results, we assume that the target task satisfies Probabilistic Lipschitzness or that is satisfies a realizability assumption (with a margin). In previous work, it is often assumed that the learner has knowledge of a hypothesis class, that "bridges" the differences between source and target tasks. Often that is a class which has small approximation error with respect to *both* source and target task (see the related work section for more details).

Summary of our results

Lower bounds: The first contribution of this chapter is a rather strong negative result: A lower bound on the sample complexity that holds for general, class, and proper DA learning. We show in Section 5.5 that even under very strong assumptions, namely, covariate shift, a bound on the point-wise weight-ratio between the two marginal distributions and realizability of the target distribution by a class of VC-dimension 1, the number of needed training examples may need to grow to infinity with the size of the domain set. (Here, "the number of needed training examples" refers to the sum of the number of source generated labeled examples and the number of target generated unlabeled examples.) In other words, even when learning from task-generated labeled examples is trivial (*one* labeled example from the target would suffice in the example constructed for our lower bound), learning based on a sample generated by a closely-related source distribution and only unlabeled target data yields very high sample complexity. We also extend our sample complexity lower bound to the case of a Euclidean domain, assuming that the labeling function is known to be Lipschitz. The bound we obtain in this situation is exponential in the Euclidean dimension of the domain. Both of our lower bounds apply regardless the choice of learning algorithm. As an aside, our lower bound employs a reduction from a novel probabilistic task that may find further applications in machine learning theory.

We complement our lower bounds with positive results for several DA settings. Our lower bound implies that the sum of the labeled source and the unlabeled target sample needs to be very large. We explore which combinations of sizes for source and target samples render DA possible. In some situations, only labeled data from a source distribution may be available. In others, there might also be unlabeled target data at hand and in this situation we would like to keep the number of labeled examples from the source rather small. We present algorithms for both these scenarios.

General DA learning: In Section 5.6, we present positive results for general DA learning (with possibly non-zero Bayes error). Assuming a bound on the weight ratio of boxes in $\mathbb{R}^d$ (or only a bound on the weight ratio of boxes of a fixed sidelength), we show that the Nearest Neighbor algorithm is a successful DA learner. Instead of using a target generated sample, one can feed the Nearest Neighbor algorithm with a sample from a source distribution (whose size depends on the usually required size for this method and the box-wise weight ratio) without worsening the error guarantee by much in comparison to Nearest Neighbor learning in the standard setting. As no data from the target task is used for learning, this constitutes a conservative DA paradigm that uses rather large amounts of labeled (source) data.

DA learning under target realizability: In Section 5.7, we present an algorithm that, instead of this large amount of labeled source data, uses unlabeled data from the target to succeed at DA. More specifically, we propose an adaptive DA paradigm for class learning under the assumption that the target distribution is realizable by some class of finite VC-dimension. Our algorithm gets a labeled source sample and then uses unlabeled target data to decide which sample points from the source sample to take into account for learning (and which ones to discard). We analyze two scenarios for this paradigm and show that here DA learning is possible on the basis of (large) unlabeled samples together with a labeled sample whose size is basically determined by the VC-dimension of the concept class (as well as a weight ratio between the two marginal distributions and the usual accuracy and confidence parameters, ϵ and δ, but does not depend on the domain size). In the first scenario we assume that the learner has prior knowledge of a concept class with respect to which the target distribution is realizable with margins. In the second scenario we assume finiteness of the domain (but no margin or Lipschitz assumptions on the involved labeling functions).

Proper DA learning: The positive result concerning Nearest Neighbor learning seem to indicate that plentiful source generated samples suffice for domain adaptation to succeed (provided some strong weight ratio assumption). However, we show in the last part of this chapter, in Section 5.8, that, in some settings, using (possibly unlabeled) target generated data is *necessary* for the success of domain adaptation learning. We consider the setting of *proper* DA learning, where the learner is required to output a predictor from some predetermined class. We show that there are cases in which, even under the assumption of a bound on the point-wise weight ratio and covariate shift, no conservative DA algorithm can succeed at the domain adaptation task. On the other hand, we present a non-conservative learning paradigm that is guaranteed to succeed in this setting. (This paradigm is similar to the two-stage learning algorithm for semi-supervised learning from the previous chapter.)

We now proceed to discussing relevant previous work on domain adaptation in Section 5.2, then set the formal framework for DA learning in Section 5.3 and present the source/target relatedness assumptions that we employ in Section 5.4.

The results of this chapter were published by Ben-David and Urner [2012] and Ben-David and Urner [2013] (and the latter publications earlier conference version by Ben-David et al. [2012]).

5.2 Related Work

Domain adaptation techniques are developed for a large variety of applications. For example, adaptation techniques are developed to account for different sources of images in object recognition tasks (Gong et al. [2012]), to make use of text documents from various languages in automated translation systems by (Daumé III and Jagarlamudi [2011]), to have an automated fatigue detection system adjust to the physiological differences in people (Chattopadhyay et al. [2011]), or to adapt from reviews from certain products to reviews for other products in sentiment analysis (Chen et al. [2011]).

The basic formal model of DA that we follow in this work was defined by Ben-David et al. [2006]. It assumes that the learner has access to a labeled sample generated by the source distribution, and an unlabeled sample from the target distribution. Below, we discuss some of the assumptions (or measures of relatedness between the source and target tasks) and methods that have been proposed to facilitate successful DA. We focus on those that are related to our work on DA.

Much of the work on DA considers the covariate shift setting, where the labeling functions of the target and source data are identical. Under this assumption, source and target data distribution only differ in their marginals. Therefore, a natural approach for covariate shift DA learning is to reweight the training sample to make it as similar as possible to a sample generated by the target distribution (see for example studies by Huang et al. [2007] and Sugiyama and Mueller [2005]). The idea is that such reweighting, when implemented precisely, turns the distribution over source generated training samples into a distribution close to that over target generated samples, thus overcoming the source-target discrepancy (since under the covariate shift assumption this discrepancy is only due to different instances having different probability weights). Similar ideas underly the methods of discrepancy minimization by Mansour et al. [2009], importance weighting by Sugiyama et al. [2007] and density ratio estimation by Tsuboi et al. [2009]. However, our sample complexity lower bound in Theorem 41 implies that in order to obtain such a reliable reweighting the learner needs access to huge samples, of sizes that go to infinity with the size of the underlying domain. (Note that *reweighting* is also referred to as *reweighing* in the literature.)

A weight ratio assumption has previously been considered by Cortes et al. [2010]. This study proposes a domain adaptation paradigm with provable success rates, assuming the learner can access the values of the point-wise weight ratio. It also acknowledges the excessive strength of an assumption that bounds the point-wise weight ratio and discuss some relaxations of this. The seeming contradiction between these sample complexity upper bounds and our lower bounds is due the the sample complexity of estimating the weight ratio (which the former analysis assumes is given to the learner). To address the sample complexity of estimating the weight-ratio function, Cortes et al. [2010] refer to Cortes et al. [2008]. However, the sample complexity analysis of Cortes et al. [2008] assumes that all the points of the labeled source sample S occur also in the unlabeled target sample T. When S and T are sampled independently, as is the case in the covariate shift DA learning setting, the size of T required to guarantee hitting every member in S grows unboundedly with the size of the support of the target distribution.

Distribution independent error bounds for domain adaptation learning were shown in an analysis of the problem with respect to a given "bridging" hypothesis class H, that is a class that is assumed to provide good approximation to *both* the source and the target distributions. Ben-David et al. [2010a] propose to measure the relatedness of the two distributions by two parameters that depend on the class H; the discrepancy between the marginal distributions by the so-called $d_{\mathcal{A}}$ distance (as introduced by Kifer et al. [2004], see Definition 37), and a notion of a "joint approximation error" of the class with respect to source and target. The paper provides upper bounds, in terms of these parameters, on the error of the simplest conservative domain adaptation algorithm—the empirical risk minimization (ERM) over the training data. A follow-up paper, Mansour et al. [2009], extends the $d_{\mathcal{A}}$ distance to real-valued function classes and loss functions other than the 0-1 loss. In addition, it proposes a non-conservative learning paradigm, a certain reweighting procedure aimed to minimize the discrepancy between the source and target input samples. This is further extended to regression problems by Cortes and Mohri [2011].

Lower bounds for DA learning under covariate shift have previously been presented by Ben-David et al. [2010b]. This study presents worst case lower bounds on the sample complexity of reweighting paradigms for DA learning in the setup of a bridging hypothesis class. Both that paper and our lower bound consider the covariate shift setup and further assume that the marginals of the training and test data are "similar". However, the notion of the distributions' similarity by Ben-David et al. [2010b], namely having small $d_{\mathcal{A}}$ distance, is weaker. The lower bound in that paper takes advantage of a looseness of the $d_{\mathcal{A}}$ distance and constructs a scenario in which, although the marginals look similar with respect to that distance, the target distribution is supported on regions that have zero weight in the training distribution. It is not surprising that under such circumstances DA may fail (the training sample misses significant parts of the target distribution). For our lower bound, we consider the strong assumption that the point-wise density ratio between the two distributions is bounded from below by 0.5 (implying that no region that is significant with respect to the target is missed by the source distribution). Nevertheless, we show that any DA algorithm may fail, even if it just has to decide between the all-zero and the all-one predictors for the target. The failure of DA in this a setting is quite surprising.

5.3 Formal Framework for Domain Adaptation

For the domain adaptation setup, where the training and test data generating distributions differ, we use the following notation: Let P^S and P^T be two distributions over $\mathcal{X} \times \{0,1\}$. We call P^S the *source distribution* and P^T the *target distribution*. We denote the marginal distribution of P^S over $\mathcal{X}$ by $P^S_{\mathcal{X}}$ and the marginal of P^T by $P^T_{\mathcal{X}}$, and their labeling rules by $l^S : \mathcal{X} \to [0,1]$ and $l^T : \mathcal{X} \to [0,1]$, respectively.

A *Domain Adaptation learner (DA learner)* takes as input a labeled *i.i.d.* sample S drawn according to P^S and an unlabeled *i.i.d.* sample T drawn according to $P^T_{\mathcal{X}}$ and aims to generate a good label predictor $h : \mathcal{X} \to \{0,1\}$ for P^T. Formally, a DA learner is a function

$$\mathcal{A} : \bigcup_{m=1}^{\infty} \bigcup_{n=1}^{\infty} ((\mathcal{X} \times \{0,1\})^m \times \mathcal{X}^n) \to \{0,1\}^{\mathcal{X}} \ .$$

We call a domain adaptation learner $\mathcal{A}$ *conservative* if it ignores the unlabeled sample it receives from the target distribution; namely, if $\mathcal{A}(U,V) = \mathcal{A}(U,W)$ for all $U \in \bigcup_{m=1}^{\infty}(\mathcal{X} \times \{0,1\})^m$ and all $V, W \in \bigcup_{n=1}^{\infty} \mathcal{X}^n$.

Definition 35 (DA Learnability)**.** Let $\mathcal{X}$ be some domain, $\mathcal{W}$ a class of pairs (P^S, P^T) of distributions over $\mathcal{X} \times \{0,1\}$, $\mathcal{C} \subseteq \{0,1\}^{\mathcal{X}}$ a hypothesis class and $\mathcal{A}$ a DA learner. We say that $\mathcal{A}$ *solves DA for* $\mathcal{C}$ *with respect to the class* $\mathcal{W}$, if there exists functions $m : (0,1) \times (0,1) \to \mathbb{N}$ and $n : (0,1) \times (0,1) \to \mathbb{N}$ such that for all pairs $(P^S, P^T) \in \mathcal{W}$, for all $\epsilon > 0$ and $\delta > 0$, when given access to a labeled sample S of size $m(\epsilon, \delta)$, generated *i.i.d.* by P^S, and an unlabeled sample T of size $n(\epsilon, \delta)$, generated i.i.d by $P^T_{\mathcal{X}}$, then, with probability at least $1 - \delta$ (over the choice of the samples S and T) $\mathcal{A}$ outputs a function h with $\mathrm{Err}_{P^T}(h) \leq \mathrm{opt}_{P^T}(\mathcal{C}) + \epsilon$. For $s \geq m(\epsilon, \delta)$ and $t \geq n(\epsilon, \delta)$, we also say that the learner $\mathcal{A}$ (ϵ, δ, s, t)*-solves DA for* $\mathcal{C}$ *with respect to the class* $\mathcal{W}$.

With this, it is easy to see how to define *general DA learning*, *class (PAC) DA learning* and *proper DA learning* analogously to Definitions 3, 5 and 6.

We are interested in finding pairs of functions m and n for the labeled source and unlabeled target samples sizes respectively, that satisfy the definitions of DA learnability for some DA learner $\mathcal{A}$. However, the preferences for either keeping the labeled source sample or the target sample small may differ in practical scenarios (according to which type of data is more readily available). Further, even for a specific DA learner, it is conceivable that the performance might be equally good when one of the two sample types is small while the other is large as when those sample sizes are switched. Thus, to not induce an unnatural emphasis on one of the two types of sample sizes, we refrain from formally defining the sample complexity of a DA learning task or that of a DA learner. For each of our positive results, we present one pair of functions m and n that attest the success of the suggested DA procedure according to the appropriate definition of DA learnability.

5.4 Relatedness Assumptions for Domain Adaptation

Clearly, the success of Domain Adaptation (DA) learning cannot be achieved for every source-target pair of learning tasks. A challenge for DA research is to discover conditions, or properties of learning tasks, that enable successful DA learning. Such properties typically express some relationship between the source and target distribution. In this section we define some such relatedness measures.

5.4.1 Covariate shift

The first property we introduce is often assumed in domain adaptation analysis (for example by Sugiyama and Mueller [2005]). In this work, we assume this property throughout.

Definition 36 (Covariate shift)**.** We say that source and target distribution satisfy the *covariate shift* property if they have the same labeling function, that is, if we have $l^S(x) = l^T(x)$ for all $x \in \mathcal{X}$. We then denote this common labeling function of P^S and P^T by l.

The covariate shift assumption is realistic for many DA tasks. For example, it is a reasonable assumption in many natural language processing (NLP) learning problems, such as part-of-speech tagging, where a learner that trains on documents from one domain is applied to a different domain. For such tasks, it is reasonable to assume that the difference between the two tasks is only in their marginal distributions over English words rather than in the tagging of each word (an adjective is an adjective independently of the type of text it occurs in). While, on first thought, it may seem like under this assumption DA becomes easy, our lower bound in Section 5.5 implies that DA remains a very hard learning problem even under covariate shift.

5.4.2 The $d_{\mathcal{A}}$-distance

DA learning may become impossible when the source and target distributions are supported on disjoint domain regions (unless the learner has prior knowledge about a mapping between the source support and the target support, in which case source and target support can be considered to be the same via this mapping). For theoretical analysis of DA learning it is thus common to put restrictions on how much the source and target weight can differ on certain subsets of the domain. We now first present the $d_{\mathcal{A}}$-distance, a measure that has first been introduced to the analysis of DA by Kifer et al. [2004], and been generalized to the notion of source-target-*discrepancy* for general loss functions (Mansour et al. [2009], Cortes and Mohri [2011]). In this work, we use this measure only in context of our lower bound.

Definition 37 ($d_{\mathcal{A}}$-Distance)**.** Let $\mathcal{X}$ be some domain, P and Q distributions over $\mathcal{X}$, and $\mathcal{A} \subseteq 2^{\mathcal{X}}$ a collection of subsets of $\mathcal{X}$ measurable with respect to both P and Q. Then the $d_{\mathcal{A}}$-distance of P and Q is defined as

$$d_{\mathcal{A}}(P, Q) = \sup_{A \in \mathcal{A}} |P(A) - Q(A)|.$$

For class learning, this measure is used as $d_{H\Delta H}$, where, for some hypothesis class $H \subseteq \{0,1\}^{\mathcal{X}}$ we define

$$H\Delta H = \{h_1 \Delta h_2 \mid h_1, h_2 \in H\},$$

and

$$h_1 \Delta h_2 = \{x \in \mathcal{X} \mid h_1(x) \neq h_2(x)\}.$$

5.4.3 Weight ratio

Another way of restricting the divergence of source and target weights on domain subsets is to assume some non-zero lower bound on the density ratio between the two distributions on these subsets. The strongest such assumption (which is nevertheless often employed) is a bound on the pointwise weight-ratio. However, this is rather unrealistic. Going back to the NLP example, it is likely that there are technical legal terms that occur in legal documents but will never show up in any news article. Furthermore, such a pointwise assumption cannot be verified from finite samples of the domain and target distributions. To overcome these drawbacks, we propose the following relaxation of that assumption.

Definition 38 (Weight ratio)**.** Let $\mathcal{B} \subseteq 2^{\mathcal{X}}$ be a collection of subsets of the domain $\mathcal{X}$ measurable with respect to both $P^S_{\mathcal{X}}$ and $P^T_{\mathcal{X}}$. For some $\eta > 0$ we define the *η-weight ratio* of the source distribution and the target distribution with respect to $\mathcal{B}$ as

$$C_{\mathcal{B},\eta}(P^S_{\mathcal{X}}, P^T_{\mathcal{X}}) = \inf_{\substack{b \in \mathcal{B} \\ P^T_{\mathcal{X}}(b) \geq \eta}} \frac{P^S_{\mathcal{X}}(b)}{P^T_{\mathcal{X}}(b)},$$

Further, we define the *weight ratio* of the source distribution and the target distribution with respect to $\mathcal{B}$ as

$$C_{\mathcal{B}}(P^S_{\mathcal{X}}, P^T_{\mathcal{X}}) = \inf_{\substack{b \in \mathcal{B} \\ P^T_{\mathcal{X}}(b) \neq 0}} \frac{P^S_{\mathcal{X}}(b)}{P^T_{\mathcal{X}}(b)}.$$

We denote the weight ratio with respect to the collection of all sets that are $P^S_{\mathcal{X}}$ and $P^T_{\mathcal{X}}$-measurable by $C(P^S_{\mathcal{X}}, P^T_{\mathcal{X}})$.

These quantities become relevant for domain adaptation when bounded away from zero. Note that in the case of discrete distributions $C(P^S_{\mathcal{X}}, P^T_{\mathcal{X}})$ is equal to the point-wise weight ratio $C(P^S_{\mathcal{X}}, P^T_{\mathcal{X}}) = C_{\{\{x\}|x\in\mathcal{X}\}}(P^S_{\mathcal{X}}, P^T_{\mathcal{X}})$. For every collection $\mathcal{B} \subseteq 2^{\mathcal{X}}$ of measurable subsets we have $C(P^S_{\mathcal{X}}, P^T_{\mathcal{X}}) \leq C_{\mathcal{B}}(P^S_{\mathcal{X}}, P^T_{\mathcal{X}})$. Thus bounding this weight ratio away from 0 is the strongest restriction.

Estimating the weight ratio from finite samples

We show that the η-weight ratio can be estimated from finite, unlabeled samples from source and target distribution for any class $\mathcal{B}$ of finite VC-dimension. For a sample S, we denote the *empirical estimate* of the weight of a set $b \in \mathcal{B}$ by $\widehat{S(b)} = \frac{|S \cap b|}{|S|}$.

For the proof of the next theorem, we need the notion of an ϵ-approximation, which was defined by Haussler and Welzl [1986].

Definition 39 (ϵ-approximation, Haussler and Welzl [1986])**.** Let $\mathcal{X}$ be some domain, $\mathcal{B} \subseteq 2^{\mathcal{X}}$ a collection of subsets of $\mathcal{X}$ and P a distribution over $\mathcal{X}$. An *ϵ-approximation* for $\mathcal{B}$ with respect to P is a finite subset $S \subseteq \mathcal{X}$ with

$$|\widehat{S(b)} - P(b)| \ \leq \ \epsilon$$

for all sets $b \in \mathcal{B}$.

It is shown in Theorem 3.3 of Haussler and Welzl [1986] that, for a collection $\mathcal{B}$ of subsets of some domain set $\mathcal{X}$ with finite VC-dimension and any distribution P over $\mathcal{X}$, an *i.i.d.* sample of size

$$\frac{16}{\epsilon^2}\left(\mathrm{VCdim}(\mathcal{B}) \ln\left(\frac{16\mathrm{VCdim}(\mathcal{B})}{\epsilon^2}\right) + \ln\left(\frac{4}{\delta}\right)\right)$$

is an ϵ-approximation for $\mathcal{B}$ with respect to P with probability at least $1 - \delta$.

Theorem 40. *Let $\mathcal{X}$ be a domain and $(P^S_{\mathcal{X}}, P^T_{\mathcal{X}})$ be a pair of (source and target) distributions over $\mathcal{X}$. Let $\mathcal{B} \subseteq 2^{\mathcal{X}}$ a collection of $P^S_{\mathcal{X}}$- and $P^T_{\mathcal{X}}$-measurable sets of finite VC-dimension. Then, for all $\epsilon > 0$ and $\delta > 0$ and $\eta > \epsilon$, with probability at least $1 - \delta$ over a sample T from $P^T_{\mathcal{X}}$ and a sample S from $P^S_{\mathcal{X}}$, each of size at least*

$$\frac{16}{\epsilon^2}\left(\mathrm{VCdim}(\mathcal{B}) \ln\left(\frac{16\mathrm{VCdim}(\mathcal{B})}{\epsilon^2}\right) + \ln\left(\frac{8}{\delta}\right)\right),$$

we have

$$\begin{aligned} C_{\mathcal{B},\eta+\epsilon}(P^S_{\mathcal{X}}, P^T_{\mathcal{X}}) &\geq \inf_{\substack{b\in\mathcal{B}\\ \widehat{T(b)}\geq\eta}} \frac{\widehat{S(b)}-\epsilon}{\widehat{T(b)}+\epsilon} \\ &\geq C_{\mathcal{B},\eta-\epsilon}(P^S_{\mathcal{X}}, P^T_{\mathcal{X}})\frac{\eta-\epsilon}{\eta+\epsilon} - \frac{2\epsilon}{\eta+\epsilon}. \end{aligned}$$

Proof. In this proof, we will use C_α as shorthand for the α-weight ratio $C_{\mathcal{B},\alpha}(P^S_{\mathcal{X}}, P^T_{\mathcal{X}})$ of $P^S_{\mathcal{X}}$ and $P^T_{\mathcal{X}}$ with respect to $\mathcal{B}$.

As discussed above, by Theorem 3.3 of Haussler and Welzl [1986], S is an ϵ-approximation for $\mathcal{B}$ with respect to $P^S_{\mathcal{X}}$ with probability at least $1-\frac{\delta}{2}$ and likewise we know that with probability at least $1-\frac{\delta}{2}$, T is an ϵ-approximation for $\mathcal{B}$ with respect to $P^T_{\mathcal{X}}$. This means that with probability at least $1-\delta$ we have

$$|\widehat{S(b)} - P^S_{\mathcal{X}}(b)| \leq \epsilon$$

and

$$|\widehat{T(b)} - P^T_{\mathcal{X}}(b)| \leq \epsilon$$

for all $b \in \mathcal{B}$. Thus we get that for all $b \in \mathcal{B}$

$$\frac{P^S_{\mathcal{X}}(b)}{P^T_{\mathcal{X}}(b)} \geq \frac{\widehat{S(b)}-\epsilon}{\widehat{T(b)}+\epsilon}.$$

With this we derive the first inequality for all $\eta > \epsilon$:

$$\begin{aligned} C_{\eta+\epsilon} &= \inf_{\substack{b\in\mathcal{B}\\ P^T(b)\geq\eta+\epsilon}} \frac{P^S_{\mathcal{X}}(b)}{P^T_{\mathcal{X}}(b)} \\ &\geq \inf_{\substack{b\in\mathcal{B}\\ P^T(b)\geq\eta+\epsilon}} \frac{\widehat{S(b)}-\epsilon}{\widehat{T(b)}+\epsilon} \\ &\geq \inf_{\substack{b\in\mathcal{B}\\ \widehat{T(b)}\geq\eta}} \frac{\widehat{S(b)}-\epsilon}{\widehat{T(b)}+\epsilon}. \end{aligned}$$

Now note that by the definition of the weight ratio, we have

$$P_{\mathcal{X}}^{S}(b) \;\geq\; C_{\alpha} P_{\mathcal{X}}^{T}(b)$$

for all $\alpha > 0$, and all $b \in \mathcal{B}$ with $P_{\mathcal{X}}^{T}(b) \geq \alpha$. As $P_{\mathcal{X}}^{T}(b) \geq \widehat{T(b)} - \epsilon$, this gives

$$P_{\mathcal{X}}^{S}(b) \;\geq\; C_{\eta}(\widehat{T(b)} - \epsilon)$$

for all b with $P_{\mathcal{X}}^{T}(b) \geq \eta - \epsilon$, which implies

$$\widehat{S(b)} \;\geq\; C_{\eta}(\widehat{T(b)} - \epsilon) - \epsilon$$

and thus, as $\widehat{T(b)} > \eta$ implies $P_{\mathcal{X}}^{T}(b) \geq \eta - \epsilon$,

$$\begin{aligned} \inf_{\substack{b \in \mathcal{B} \\ \widehat{T(b)} \geq \eta}} \frac{\widehat{S(b)} - \epsilon}{\widehat{T(b)} + \epsilon} \;&\geq\; \inf_{\substack{b \in \mathcal{B} \\ \widehat{T(b)} \geq \eta}} \frac{C_{\eta-\epsilon}(\widehat{T(b)} - \epsilon) - 2\epsilon}{\widehat{T(b)} + \epsilon} \\ &\geq\; \frac{C_{\eta-\epsilon}(\eta - \epsilon) - 2\epsilon}{\eta + \epsilon}, \end{aligned}$$

where the last inequality holds as the expression

$$\frac{C_{\eta-\epsilon}(\widehat{T(b)} - \epsilon) - 2\epsilon}{\widehat{T(b)} + \epsilon}$$

is monotonously increasing in $\widehat{T(b)}$ and $\widehat{T(b)} \geq \eta$. □

This theorem quantifies the information that the empirical estimate of the η-weight ratio provides about the true η-weight ratio. The first inequality tells us that if the ϵ-estimate (the middle term in the theorem) of this weight ratio is large, then the true weight ratio is also large (for slightly larger η). On the other hand, the second inequality shows that, if the true weight ratio is large, then the empirical estimate (for a slightly larger η) is also bounded away from 0. Note that, if $\epsilon \leq \frac{C_{\eta-\epsilon}\eta}{2+C_{\eta-\epsilon}}$, then $\frac{C_{\eta-\epsilon}(\eta-\epsilon)-2\epsilon}{\eta+\epsilon} > 0$, and this condition holds for sufficiently small ϵ.

The theorem further implies that, if source and target distribution are absolutely continuous (have a density function), then with growing sizes of source and target samples S and T, the empirical weight ratio $\frac{\widehat{S(b)}}{\widehat{T(b)}}$ of subsets $b \in \mathcal{B}$ of target weight at least η converges to $C_{\mathcal{B},\eta}(P^S_{\mathcal{X}}, P^T_{\mathcal{X}})$: As ϵ tends to 0, both

$$C_{\mathcal{B},\eta+\epsilon}(P^S_{\mathcal{X}}, P^T_{\mathcal{X}})$$

and

$$C_{\mathcal{B},\eta-\epsilon}(P^S_{\mathcal{X}}, P^T_{\mathcal{X}})\frac{\eta-\epsilon}{\eta+\epsilon} - \frac{\epsilon}{\eta+\epsilon}$$

converge to $C_{\mathcal{B},\eta}(P^S_{\mathcal{X}}, P^T_{\mathcal{X}})$, and thus so does

$$\inf_{\substack{b \in \mathcal{B} \\ \widehat{T(b)} \geq \eta}} \frac{\widehat{S(b)} - \epsilon}{\widehat{T(b)} + \epsilon}.$$

In contrast with the above results, the negative result of Theorem 41 implies that any stronger weight ratio estimate that suffices for domain adaptation learning cannot be reliably obtained from finite samples (of distribution-independent sizes). In particular, this negative result applies to estimating the weight ratio at every point (as opposed to the η-weight ratio), since it is easy to see that Domain Adaptation under covariate shift and a non-zero lower bound of 1/2 on the point-wise weight ratio, can be readily done upon having a good estimate of the weight ratio at every point.

5.5 Lower Bounds for Realizable Domain Adaptation

The lower bound in this section shows that no reasonably small amount of labeled source and unlabeled target data suffices for DA under covariate shift even with a large bound on the pointwise weight ratio. We show that even in the case where the learner knows that the target is realizable by the class $H_{1,0}$ that contains only the all-1 and the all-0 labeling functions, a class of VC-dimension 1, the sizes of the source sample and the target sample need to be (roughly) as large as the square-root of the domain size for domain adaptation to be possible. This is in sharp contrast to the standard learning framework, where realizability by a small class represents a very strong prior knowledge, that renders the learning task very easy. In fact, realizability by the class $H_{1,0}$ implies that one labeled example would suffice for standard learning.

Theorem 41. *For every finite domain $\mathcal{X}$, for every ϵ and δ with $\epsilon + \delta < 1/2$, no algorithm can (ϵ, δ, s, t)-solve the DA problem for the class $\mathcal{W}$ of pairs (P^S, P^T) satisfying covariate shift, with $C(P^S, P^T) \geq 1/2$, $d_{H_{1,0}\Delta H_{1,0}}(P^S, P^T) = 0$ and $\mathrm{opt}_{P_T}(H_{1,0}) = 0$ if*

$$s + t < \min\{\sqrt{\ln(2)|\mathcal{X}|}, \sqrt{\ln(1/(2(\epsilon + \delta)))|\mathcal{X}|}\} - 1.$$

This lower bound implies that even for relatively large values of ϵ and δ, namely any ϵ and δ with $\epsilon + \delta \leq 1/4$, the required sample size for DA is lower bounded by $\sqrt{\ln(2)|\mathcal{X}|}$ and thus grows with the domain size. The proof of this theorem is presented in Section 5.5.1.

This hardness result is quite surprising since it applies to a setting in which DA learning is seemingly as easy as it can get; the prior knowledge about the target task is so strong that one labeled target example would suffice for finding a zero error classifier. Furthermore, the source and target distributions share the same deterministic labeling function, and the marginals of the two distributions are similar from both the $d_{H\Delta H}$-distance and the weight ratio perspectives (namely the source probability of any domain point is at least half its target probability).

Several conclusions can be drawn from this lower bound:

1. If one assumes target realizability by a small hypothesis class but does not assume that there is such a class that has small approximation error with respect to both the source and the target, the DA sample complexity cannot be bounded as a function of only the VC-dimension of the class that realizes the target distribution. This is in sharp contrast to the sample complexity of standard learning without discrepancy between the training and test data.

2. It is necessary to have some data generated by the target distribution available, if the number of labeled examples is only allowed to depend on the VC-dimension of the class.

3. Since having access to the ratio between a the source and target probability of domain points allows successful DA learning under the covariate shift assumption and bounded weight ratio, our result implies that the sample sizes needed to obtain useful approximations of that ratio, as required, e.g. , for importance weighting techniques, are prohibitively high.

A lower bound in terms of Lipschitzness Our next result, Theorem 42, implies a lower bound for the size of the sample for infinite domains under the additional assumption that the labeling function satisfies the Lipschitz property for some Lipschitz constant L. Again, this lower bound holds under the assumption that the target is realizable by a two-function-class.

Theorem 42. *Let $\mathcal{X} = [0,1]^d$, $\epsilon > 0$ and $\delta > 0$ be such that $\epsilon + \delta < 1/2$, let $L > 1$ and let $\mathcal{W}_L$ be the set of pairs (P^S, P^T) of distributions over $\mathcal{X}$ satisfying covariate-shift, with $\mathrm{opt}_{P_T}(H_{1,0}) = 0$, $C(P^S, P^T) \geq 1/2$, $d_{H_{1,0}\Delta H_{1,0}}(P^S, P^T) = 0$ and L-Lipschitz labeling functions l. Then no DA-learner can (s, t, ϵ, δ)-solve the DA-problem for the class $\mathcal{W}_L$ unless*

$$s + t < \min\{\sqrt{\ln(2)(L+1)^d}, \sqrt{\ln(1/(2(\epsilon+\delta)))(L+1)^d}\} - 1.$$

Proof. Let $\mathcal{G} \subseteq \mathcal{X}$ be the points of a grid in $[0,1]^d$ with distance $1/L$. Then we have $|\mathcal{G}| = (L+1)^d$. The class $\mathcal{W}_L$ contains all triples (P^S, P^T, l), where the support of P^S and P^T is $\mathcal{G}$, $\mathrm{opt}_{P_T}(H_{1,0}) = 0$, $C(P^S, P^T) \geq 1/2$, $d_{H_{1,0}\Delta H_{1,0}}(P^S, P^T) = 0$ and arbitrary labeling functions $l : \mathcal{G} \to \{0,1\}$, as every such function is L-Lipschitz. As $\mathcal{G}$ is finite, the bound follows from Theorem 41. □

5.5.1 Proof of Theorem 41

This section is devoted to the proof of our main lower bound in Theorem 41. We start by presenting the Left/Right problem, a statistical decision problem that we make use of, and provide an overview on the idea of the lower bound proof. We then derive a lower bound on the needed sample sizes to solve the Left/Right problem and, in the last part of this section, proceed to a reduction of the Left/Right problem to DA, which establishes the hardness of DA.

The Left/Right problem

We consider the problem of distinguishing two distributions from finite samples. The Left/Right Problem was introduced by Kelly et al. [2010]:

Input Three finite samples, L, R and M of points from some domain set $\mathcal{X}$.

Question Assuming that L is an an *i.i.d.* sample from some distribution P over $\mathcal{X}$, that R is an an *i.i.d.* sample from some distribution Q over $\mathcal{X}$, and that M is an *i.i.d.* sample generated by one of these two probability distributions, was M generated by P or by Q ?

Overview on the proof idea

We first derive a lower bound on the sample size needed to solve the Left/Right problem in Lemma 44. Then we reduce the Left/Right problem to the problem of domain adaptation under target realizability, thereby obtaining a lower bound on the sample size needed to solve DA. Intuitively, one can not answer the Left/Right-question if the sample M intersects neither the sample L nor the sample R. This yields a lower bound for the Left/Right problem in the order of the square-root of the domain size by a standard calculation from the “Birthday paradox”. The idea of the reduction to domain adaptation is to define a source distribution that is a balanced mixture of P and Q with a labeling function that gives label 1 to points from L (generated by P) and label 0 to points from R (generated by Q). The sample M can then be considered an unlabeled sample from a target distribution that is equal to either P or Q. Thus, predicting label 0 or 1 correctly corresponds to deciding whether M was generated by P or by Q. Thereby, we obtain a lower bound for domain adaptation for the sum of the sizes of the labeled source sample and the unlabeled target sample, in the order of the square-root of the domain size.

Lower bound for the Left/Right problem

We use the following notion of success for an algorithm on the Left/Right problem:

Definition 43 (Left/Right problem solvability)**.** We say that a (randomized) algorithm (δ, l, r, m)-*solves the Left/Right problem* with respect to a class $\mathcal{W}$ of triples (P_1, P_2, P_3) of distributions (where $P_3 = P_1$ or $P_3 = P_2$), if, given samples L *i.i.d.* from P_1, R *i.i.d.* from P_2 and M *i.i.d.* from P_3 of sizes l, r and m respectively, it correctly decides whether $P_3 = P_1$ or $P_3 = P_2$ with probability at least $1 - \delta$.

We will now show that for any sample sizes l, r and m and for any $\gamma < 1/2$, there exists a finite domain $\mathcal{X} = \{1, 2, \ldots, n\}$ and a small class $\mathcal{W}_n^{uni}$ of triples of distributions over $\mathcal{X}$ such that no algorithm can (γ, l, r, m)-solve the Left/Right problem with respect to this class. In our class, both the distribution generating L and the distribution generating R are uniform over half of the points in $\mathcal{X}$, but their supports are disjoint. Formally, we construct the class as follows: $\mathcal{W}_n^{uni} = \{(U_A, U_B, U_C) \mid A \cup B = \{1, \ldots n\}, A \cap B = \emptyset,\ |A| = |B|,\ \text{and } C = A \text{ or } C = B\}$, where, for a finite set Y, U_Y denotes the uniform distribution over Y. With this we obtain:

Lemma 44. (Left/Right problem lower bound) *For any given sample sizes l for L, r for R and m for M and any $0 < \gamma < 1/2$, if $k = \max\{l, r\} + m$, then for*

$$n > \max\{k^2/\ln(2), k^2/\ln(1/2\gamma)\}$$

no algorithm has probability of success greater than $1 - \gamma$ over the class $\mathcal{W}_n^{uni}$.

The rest of this subsection is devoted to prove this lemma. For this, we employ a method introduced by Batu et al. [2013] in the context of deriving a lower bound on the sample size for a related problem. The authors show that, when testing so-called permutation invariant properties, that is, if the property does not change with permuting the underlying domain, it suffices to consider algorithms that take only a *fingerprint* of the sample as input (see precise definition below). Note that the Left/Right problem is permutation-invariant, since, whether M is a sample from P or from Q does not depend on a permutation of $\mathcal{X}$.

Definition 45 (Fingerprint)**.** Let L, R, M be three multi-sets of sizes at most n each sampled from distributions P or Q over some domain $\mathcal{X}$ as in the definition of the Left/Right problem. We define the *fingerprint* of this triple of multi sets as the set $\{C_{i,j,k} \mid 1 \leq i, j, k \leq n\}$ where $C_{i,j,k}$ is the number of elements of $\mathcal{X}$, that appear exactly i times in L, j times in R and k times in M.

The following lemma allows us to restrict our attention to fingerprints of an instance of the Left/Right problem as input.

Lemma 46 (Batu et al. [2013])**.** *If there exists an algorithm $\mathcal{A}$ for testing some permutation-invariant property of distributions, then there exists an algorithm for that same task that gets as input only the fingerprints of the samples that $\mathcal{A}$ takes and enjoys the same guarantee on its probability of success.*

Proof sketch. This lemma is proven by Batu et al. by showing how to reconstruct the samples from a fingerprint for some fixed permutation of the distribution. To see this, one notes that each element of $\mathcal{X}$ contributes to at most one of the $C_{i,j,k}$. Thus, an algorithm can reconstruct a permuted sample from the fingerprint and then feed this sample as input to $\mathcal{A}$. As the property is permutation-invariant, this can not change the (distribution over the) output(s). □

The following lemma gives a lower bound on the sample size needed to see repetitions in a sample from a uniform distribution over a finite domain. Its proof is similar to the calculations used for establishing the no-collision probabilities in the well known "Birthday paradox". We include this calculation for completeness.

Lemma 47. *Let $\mathcal{X}$ be a finite domain of size m. For every $0 < \beta < 1$, with probability exceeding β, an i.i.d. sample of size at most $n \leq \min\{\sqrt{\ln(2)m}, \sqrt{\ln(1/\beta)m}\}$ uniformly drawn over $\mathcal{X}$, contains no repeated elements.*

Proof. Note that, we always have $n \leq m/2$. Now, the probability of not drawing any element twice, when choosing n elements uniformly at random from a set of size m is

$$\begin{aligned} & \frac{m-1}{m} \cdot \frac{m-2}{m} \cdot \ldots \cdot \frac{m-(n-1)}{m} \\ = \; & (1-\frac{1}{m}) \cdot (1-\frac{2}{m}) \cdot \ldots \cdot (1-\frac{n-1}{m}) \\ \geq \; & \mathrm{e}^{-2(\frac{1}{m}+\frac{2}{m}+\ldots+\frac{(n-1)}{m})} \\ = \; & \mathrm{e}^{-\frac{n(n-1)}{m}} \\ \geq \; & \mathrm{e}^{-\frac{n^2}{m}}, \end{aligned}$$

where the first inequality holds as $(1-x) \geq \mathrm{e}^{-2x}$ for $0 \leq x \leq 1/2$. Thus, if

$$n \leq \min\{\sqrt{\ln(2)m}, \sqrt{\ln(1/\beta)m}\},$$

then we get that this probability is at least β (for $\beta \leq 1/2$, it is at least $1/2$, and for $\beta \geq 1/2$ it is at least β). □

With this, we can now complete the proof of Lemma 44.

Proof of Lemma 44. Set $\beta = 2\gamma$. By Lemma 47, with probability exceeding β the input to the Left/Right problem over $\mathcal{W}_n^{uni}$ has no repeated elements and the three input samples are disjoint. Consequently, with probability exceeding β, the fingerprint F of the input has $C_{1,0,0} = l$, $C_{0,1,0} = r$, $C_{0,0,1} = m$ and $C_{i,j,k} = 0$ for all other combinations of i, j and k independently of whether the sample M was generated by U_A or by U_B.

Let $\mathcal{A}$ be some algorithm. Let $p \in [0,1]$ be the probability that $\mathcal{A}$ outputs U_A on input F. Now, if $p \leq 1/2$ we have that $\mathcal{A}$ errs with probability larger than $\beta/2 = \gamma$ for all triples where C is equal to B. Otherwise it errs with probability larger than $\beta/2 = \gamma$ on all triples where C is equal to A. Thus, no algorithm can (γ, l, r, m)-solve the Left/Right problem for the class $\mathcal{W}_n^{uni}$. □

Reducing the Left/Right problem to domain adaptation learning

In order to reduce the Left/Right problem to domain adaptation, we define a class of DA problems that corresponds to the class of triples $\mathcal{W}_n^{uni}$, for which we have proven a lower bound on the sample sizes needed for solving the Left/Right problem. For a number n, let $\mathcal{W}_n^{DA}$ be the class of pairs (P^S, P^T), where $P^S_{\mathcal{X}}$ is uniform over some finite set $\mathcal{X}$ of size n, $P^T_{\mathcal{X}}$ is uniform over some subset U of $\mathcal{X}$ of size $n/2$ and their common labeling function l assigns points in U to 1 and points in $\mathcal{X} \setminus U$ to 0 or vice versa. Note that we have $C(P^S, P^T) = 1/2$ and $d_{H_{1,0}\Delta H_{1,0}}(P^S, P^T) = 0$ for all (P^S, P^T) in $\mathcal{W}_n^{DA}$. Further, for the class $H_{1,0}$ that contains only the constant 1 function and the constant 0 function, we have $\mathrm{opt}_{P_T}(H_{1,0}) = 0$ for all elements of $\mathcal{W}_n^{DA}$.

Lemma 48. *The Left/Right problem reduces to domain adaptation. More precisely, given a number n and an algorithm $\mathcal{A}$ that, given the promise that the target task is realizable by the class $H_{1,0}$, can (ϵ, δ, s, t)-solve DA for a class $\mathcal{W}$ that includes $\mathcal{W}_n^{DA}$, we can construct an algorithm that $(\epsilon + \delta, s, s, t + 1)$-solves the Left/Right problem on $\mathcal{W}_n^{uni}$.*

Proof. Assume we are given samples $L = \{l_1, l_2, \ldots, l_s\}$ and $R = \{r_1, r_2, \ldots, r_s\}$ of size s and a sample M of size $t+1$ for the Left/Right problem coming from a triple (U_A, U_B, U_C) of distributions in $\mathcal{W}_n^{uni}$. We construct an input to domain adaptation by setting the unlabeled target sample $T = M \setminus \{p\}$ where p is a point from M chosen uniformly at random and construct the labeled source sample S as follows: We select s elements from $L \times \{0\} \cup R \times \{1\}$ by successively flipping an unbiased coin, and depending on the output choosing the next element from $L \times \{0\}$ or $R \times \{1\}$.

These sets can now be considered as an input to domain adaptation generated from a source distribution $P^S = U_{A \cup B}$ that is uniform over $A \cup B$. The target distribution P^T of this domain adaptation instance has marginal equal to U_A or to U_B (depending on whether M was a sample from U_A or from U_B). The labeling function of this domain adaptation instance is $l(x) = 0$ if $x \in A$ and $l(x) = 1$ if $x \in B$. Observe that we have $C(P^S, P^T) = 1/2$, $\mathrm{opt}_{P_T}(H_{1,0}) = 0$, and $(P^S, P^T) \in \mathcal{W}_n^{DA}$. Assume that h is the output of $\mathcal{A}$ on input S and T. The algorithm for the Left/Right problem then outputs U_A if $h(p) = 0$ and U_B if $h(p) = 1$ and the claim follows as we have $\mathrm{Err}_h(P^S) \leq \epsilon$ with confidence $1 - \delta$. □

Lemma 44 together with Lemma 48 shows that no algorithm can solve the DA problem for $\mathcal{W}_n^{DA}$, even under the assumption of realizability by $H_{1,0}$, if the sample sizes of the source and target sample satisfy $|S| + |T| + 1 < \min\{\sqrt{\ln(2)|\mathcal{X}|}, \sqrt{\ln(1/(2(\epsilon + \delta)))|\mathcal{X}|}\}$. This completes the proof of Theorem 41.

5.6 General DA Learning

We now turn to presenting algorithms that do succeed at domain adaptation learning. In this section, we show that for the Nearest Neighbor algorithm a target generated sample can be replaced by a source generated sample while maintaining the error guarantee. For this, we employ a Lipschitzness assumption on the labeling function and a weight-ratio assumption with respect to the class of axis-aligned rectangles. Note that for a fixed dimension of the space, this class is of finite VC-dimension and the η-weight ratio can therefore be estimated from finite samples as shown in section 5.4.3.

We consider the target task performance of a Nearest Neighbor method with respect to a source labeled training sample. Given a labeled sample $S \subseteq \mathcal{X} \times \{0,1\}$, this Nearest Neighbor algorithm outputs a function $\mathrm{NN}(S)$ that assigns to each point the label of its nearest neighbor in the sample S. We will analyze its performance as a function of the Lipschitzness and the weight ratio.

We use the following notation: Let $S_{\mathcal{X}}$ denote the sample points of S without labels (namely, $S_{\mathcal{X}} := \{x \in \mathcal{X} \mid \exists y \in \{0,1\} : (x,y) \in S\}$). For any $x \in S_{\mathcal{X}}$, let $l^S(x)$ denote the label of the point x in the sample S. Given some labeled sample set S and a point $x \in \mathcal{X}$, let $N_S(x)$ denote the nearest neighbor to x in S, $N_S(x) = \mathrm{argmin}_{z \in S_{\mathcal{X}}} \|x - z\|$. We define $\mathrm{NN}(S)$ for all points $x \in \mathcal{X}$ by $\mathrm{NN}(S)(x) = l^S(N_S(x))$ (as in Definition 11).

In this chapter, we assume that our domain is the unit cube $\mathcal{X} = [0,1]^d$. We let $\mathcal{B}$ denote the set of axis alligned rectangles in $[0,1]^d$ and $\mathcal{B}_\gamma$ the class of axis aligned rectangles with sidelength γ for some $\gamma > 0$. We will analyze the Nearest Neighbor algorithm for domain adaptation under Probabilistic Lipschitzness with a bound on either $\mathcal{B}_\gamma$, for some fixed γ, or on $\mathcal{B}$. For both settings, we consider the case that the labeling function is deterministic as well as having a probabilistic labeling function.

Note that the exponential dependence on the dimension of the space in the bounds of this section is inevitable. This follows, for example, from our lower bound in Theorem 42. But, as standard learning can be viewed as a special case of domain adaptation learning (where source and target are the same), this also already follows from the lower bound in Theorem 14 (see Chapter 2) and is known as "the curse of dimensionality of Nearest

Neighbor learning" (see discussion in Section 19.2.2 of Shalev-Shwartz and Ben-David [2014]).

We start by giving a bound for deterministic labeling ($l : \mathcal{X} \to \{0, 1\}$) when the weight ratio is bounded for $\mathcal{B}_\gamma$. The proofs of the following two theorems are adaptations of the proof of Theorem 20 to the DA setting.

Theorem 49. *Let the domain be the unit cube, $\mathcal{X} = [0, 1]^d$, and for some $C > 0$ and some $\gamma \geq 0$, let $\mathcal{W}$ be a class of pairs (P^S, P^T) of source and target distributions over $\mathcal{X} \times \{0, 1\}$ satisfying the covariate shift assumption, with $C_{\mathcal{B}_\gamma}(P^S_{\mathcal{X}}, P^T_{\mathcal{X}}) \geq C$, and their common deterministic labeling function $l : \mathcal{X} \to \{0, 1\}$ satisfying the ϕ-probabilistic-Lipschitz property with respect to the target distribution, for some function ϕ. Then, for all m, and all $(P_S, P_T) \in \mathcal{W}$,*

$$\mathbb{E}_{S \sim P^{S^m}}[\mathrm{Err}_{P^T}(h_{\mathrm{NN}})] \leq \phi\left(\gamma\sqrt{d}\right) + \frac{1}{\gamma^d\, m\, \mathrm{e}\, C}.$$

Proof. Consider a cover of the space $[0, 1]^d$ with boxes of sidelength γ. A test point from the target gets the wrong label only if it sits in a box that was not hit by the source sample S or if its box contains a point of S but that point from S has the opposite label.

The latter probability is bounded by $\phi\left(\gamma\sqrt{d}\right)$: A box of side-length γ has diameter $\gamma\sqrt{d}$ and thus

$$\begin{aligned} & \Pr_{x \sim P^T}[l(x) \neq l(N_S(x)) \wedge \|N_S(x) - x\| \leq \gamma\sqrt{d}] \\ \leq \ & \Pr_{x \sim P^T}[|l(x) - l(N_S(x))| \geq (1/\gamma\sqrt{d}) \cdot \|N_S(x) - x\|] \\ \leq \ & \phi\left(\gamma\sqrt{d}\right). \end{aligned}$$

The probability that a point is in a box that is not hit by S can be bounded using Lemma 19 (see Chapter 3). We can cover the space with $(1/\gamma^d)$ many boxes of sidelength γ. Let $C_1, C_2, \ldots, C_{1/\gamma^d}$ be such a cover of the set $[0, 1]^d$ using boxes of side-length γ. We have

$$P^T_{\mathcal{X}}(C_i) \leq \frac{1}{C_{\mathcal{B}}(P^S_{\mathcal{X}}, P^T_{\mathcal{X}})} P^S_{\mathcal{X}}(C_i) \leq \frac{1}{C} P^S_{\mathcal{X}}(C_i)$$

for all these boxes C_i. Thus, Lemma 19 yields

$$\begin{aligned}
&\mathbb{E}_{S\sim P_{\mathcal{X}}^{S\,m}}\left[\sum_{i:C_i\cap S=\emptyset} P_{\mathcal{X}}^T[C_i]\right] \\
\leq\ &\mathbb{E}_{S\sim P_{\mathcal{X}}^{S\,m}}\left[\sum_{i:C_i\cap S=\emptyset} \frac{1}{C} P_{\mathcal{X}}^S[C_i]\right] \\
\leq\ &\frac{1}{C}\,\mathbb{E}_{S\sim P_{\mathcal{X}}^{S\,m}}\left[\sum_{i:C_i\cap S=\emptyset} P_{\mathcal{X}}^S[C_i]\right] \ \leq\ \frac{1}{\gamma^d\, C\, m\, \mathrm{e}}.
\end{aligned}$$

The two bounds together imply the theorem. □

If, for the case of deterministic labeling, the weight ratio is bounded for the set $\mathcal{B}$ of *all* axis alligned rectangles, we get the following result of DA learnability:

Theorem 50. *Let the domain be the unit cube, $\mathcal{X} = [0,1]^d$, and for some $C > 0$, let $\mathcal{W}$ be a class of pairs (P^S, P^T) of source and target distributions over $\mathcal{X} \times \{0,1\}$ satisfying the covariate shift assumption, with $C_{\mathcal{B}}(P_{\mathcal{X}}^S, P_{\mathcal{X}}^T) \geq C$, and their common deterministic labeling function $l : \mathcal{X} \to \{0,1\}$ satisfying the ϕ-probabilistic-Lipschitz property with respect to the target distribution, for some function ϕ. Then, for all $\epsilon, \delta > 0$, for all $(P^S, P^T) \in \mathcal{W}$, if S is a source generated sample of size at least*

$$m \ \geq\ \frac{2}{\epsilon\,\delta\, C\,\mathrm{e}}\left(\frac{\sqrt{d}}{\phi^{-1}(\epsilon/2)}\right)^d,$$

then, with probability at least $1-\delta$ (over the choice of S), the target error of the Nearest Neighbor classifier is at most ϵ.

Proof. As shown in the proof of the previous theorem, we can bound the total target weight of boxes of some sidelength γ that are not hit by a source sample S by

$$\mathbb{E}_{S\sim P_{\mathcal{X}}^{S\,m}}\left[\sum_{i:C_i\cap S=\emptyset} P_{\mathcal{X}}^T[C_i]\right] \ \leq\ \frac{1}{\gamma^d\, C\, m\, \mathrm{e}}.$$

Applying Markov's inequality thus yields

$$\Pr_{S\sim P_{\mathcal{X}}^{S^m}}\left[\sum_{i:C_i\cap S=\emptyset} P_{\mathcal{X}}^T[C_i] \;\geq\; \epsilon/2\right] \;\leq\; \frac{2}{\epsilon\,\gamma^d\,C\,m\,\mathrm{e}}.$$

Setting this inequality to be smaller than δ yields that if $m \geq \frac{2}{\epsilon\delta\gamma^d C\mathrm{e}}$, then with probability at least $1-\delta$ at most an $\epsilon/2$-fraction of points are sitting in boxes that are not hit by the sample. The error in this case is thus bounded by $\epsilon/2 + \phi(\gamma\sqrt{d})$. Now, setting $\gamma = \phi^{-1}(\epsilon/2)/\sqrt{d}$ implies the claimed statement. □

We now give a similar bound for general labeling functions $l : \mathcal{X} \to [0,1]$. The proof of this theorem is an adaptation of the Nearest Neighbor analysis in Shalev-Shwartz and Ben-David [2014] to the case of domain adaptation and Probabilistic Lipschitzness. We start with the case of a bounded weight ratio for $\mathcal{B}_\gamma$.

Theorem 51. *Let the domain be the unit cube, $\mathcal{X} = [0,1]^d$, and for some $C > 0$ and $\gamma \geq 0$, let $\mathcal{W}$ be a class of pairs (P^S, P^T) of source and target distributions over $\mathcal{X} \times \{0,1\}$ satisfying the covariate shift assumption, with $C_{\mathcal{B}_\gamma}(P_{\mathcal{X}}^S, P_{\mathcal{X}}^T) \geq C$, and their common labeling function $l : \mathcal{X} \to [0,1]$ satisfying the ϕ-probabilistic-Lipschitz property with respect to the target distribution, for some function ϕ. Then, for all λ and all m, we have*

$$\mathbb{E}_{S\sim P^{S^m}}[\mathrm{Err}_{P^T}(h_{\mathrm{NN}})] \leq 2\,\mathrm{opt}_{P^T} + \phi(\lambda) + \frac{\sqrt{d}}{\lambda\,C}\left(\frac{1}{m\,\mathrm{e}\,\gamma^d} + \gamma\right).$$

Proof. We start by proving that

$$\begin{aligned} &\mathbb{E}_{S\sim P^{S^m}}[\mathrm{Err}_{P^T}(h_{NN})] && (5.1)\\ \leq\;& 2\,\mathrm{opt}_{P^T} + \phi(\lambda) + \lambda^{-1}\mathbb{E}_{S\sim P^{S^m}, x\sim P_{\mathcal{X}}^T}[\|x - N_S(x)\|] && (5.2)\end{aligned}$$

We first note that given two instances x, x' we have

$$\begin{aligned}\Pr_{y\sim l(x), y'\sim l(x')}[y \neq y'] \;&=\; l(x)(1-l(x')) + l(x')(1-l(x))\\ &\leq\; 2l(x)(1-l(x)) + |l(x') - l(x)|\ .\end{aligned}$$

The error of the NN procedure can be therefore written as

$$\begin{aligned}
& \mathbb{E}_{S\sim P^{S^m}}[\mathrm{Err}_{P^T}(h_{\mathrm{NN}})] \\
= \; & \mathbb{E}_{S\sim P^{S^m}}\mathbb{E}_{x\sim P^T}\left[\Pr_{y\sim l(x),y'\sim l(N_S(x))}[y\neq y']\right] \\
\leq \; & \mathbb{E}_{S\sim P^{S^m}}\mathbb{E}_{x\sim P^T}\left[2l(x)(1-l(x))+|l(N_S(x))-l(x)|\right] \\
\leq \; & 2\,\mathrm{opt}_{P^T}+\mathbb{E}_{S\sim P^{S^m}}\mathbb{E}_{x\sim P^T}[|l(N_S(x))-l(x)|] \; .
\end{aligned}$$

Using the definition of Probabilistic Lipschitzness and the fact that the range of l is $[0,1]$, we have, for any set S,

$$\mathbb{E}_{x\sim P^T}[|l(N_S(x))-l(x)|]\leq \phi(\lambda)+\lambda^{-1}\mathbb{E}_{x\sim P^T}[\|N_S(x)-x\|] \; ,$$

which yields equation 5.1. Thus, in order to prove learnability, we need an upper bound on

$$\mathbb{E}_{S\sim P^{S^m},x\sim P^T_{\mathcal{X}}}[\|x-N_S(x)\|].$$

Now, let $C_1,\ldots,C_r$ be the cover of the set $[0,1]^d$ using boxes of side-length γ. We have

$$P^T_{\mathcal{X}}(C_i)\leq \frac{1}{C_{\mathcal{B}_\gamma}(P^S_{\mathcal{X}},P^T_{\mathcal{X}})}P^S_{\mathcal{X}}(C_i)\leq \frac{1}{C}P^S_{\mathcal{X}}(C_i)$$

for all boxes C_i. Thus, Lemma 19 yields

$$\begin{aligned}
\mathbb{E}_{S\sim P^{S^m}_{\mathcal{X}}}\left[\sum_{i:C_i\cap S=\emptyset}P^T_{\mathcal{X}}[C_i]\right] \; & \leq \; \mathbb{E}_{S\sim P^{S^m}_{\mathcal{X}}}\left[\sum_{i:C_i\cap S=\emptyset}\frac{1}{C}P^S_{\mathcal{X}}[C_i]\right] \\
& \leq \; \frac{1}{C}\mathbb{E}_{S\sim P^{S^m}_{\mathcal{X}}}\left[\sum_{i:C_i\cap S=\emptyset}P^S_{\mathcal{X}}[C_i]\right] \; \leq \; \frac{r}{Cme}
\end{aligned}$$

For each x,x' in the same box we have $\|x-x'\|\leq\sqrt{d}\,\gamma$. Otherwise, $\|x-x'\|\leq\sqrt{d}$. For $x\in\mathcal{X}$ we let $C_x\in\{C_1,\ldots,C_r\}$ denote the box that contains the point x. Therefore,

$$\begin{aligned}
& \mathbb{E}_{S\sim P^{S^m},x\sim P^T_{\mathcal{X}}}[\;\|x-N_S(x)\|\;] \\
\leq \; & \mathbb{E}_{S\sim P^{S^m}}\left[\Pr_{x\sim P^T_{\mathcal{X}}}[C_x\cap S=\emptyset]\sqrt{d}+\Pr_{x\sim P^T_{\mathcal{X}}}[C_x\cap S\neq\emptyset]\sqrt{d}\,\gamma\right] \\
\leq \; & \sqrt{d}(\frac{r}{me\,C}+\gamma) \; .
\end{aligned}$$

Since the number of boxes is $(1/\gamma)^d$ we get that

$$\mathbb{E}_{S\sim P^{S^m}, x\sim P^T_{\mathcal{X}}}[\|x - N_S(x)\|] \leq \sqrt{d}\left(\frac{1}{m\,\mathrm{e}\,C\,\gamma^d} + \gamma\right) \ .$$

Combining this with equation 5.1, we get

$$\begin{array}{rl} & \mathbb{E}_{S\sim P^{S^m}}\left[\mathrm{Err}_{P^T}(h_{\mathrm{NN}})\right] \\ \leq & 2\,\mathrm{opt}_{P^T} + \phi(\lambda) + \lambda^{-1}\sqrt{d}\left(\frac{1}{m\,\mathrm{e}\,C\,\gamma^d} + \gamma\right) \\ \leq & 2\,\mathrm{opt}_{P^T} + \phi(\lambda) + \lambda^{-1}\sqrt{d}\,\frac{1}{C}\left(\frac{1}{m\,\mathrm{e}\,\gamma^d} + \gamma\right) \end{array}$$

□

Note that, in the setting of the above theorem, for $m \geq \frac{2}{\gamma^{d+1}}$, we get, for all $\lambda > 0$ that

$$\mathbb{E}_{S\sim P^{S^m}}[\mathrm{Err}_{P^T}(h_{\mathrm{NN}})] \leq 2\,\mathrm{opt}_{P^T} + \phi(\lambda) + 2\lambda^{-1}\frac{\sqrt{d}}{C}\gamma \ .$$

With a bound on the weight ratio for the set $\mathcal{B}$ of all axis alligned rectangles we get an error bound that approaches $2\,\mathrm{opt}_{P^T} + \phi(\lambda)$ as the sample size grows.

Theorem 52. *Let our domain $\mathcal{X}$ be the unit cube, $[0,1]^d$, and for some $C > 0$, let $\mathcal{W}$ be a class of pairs (P^S, P^T) of source and target distributions over $\mathcal{X} \times \{0,1\}$ satisfying the covariate shift assumption, with $C_{\mathcal{B}}(P^S_{\mathcal{X}}, P^T_{\mathcal{X}}) \geq C$, and their common labeling function $l : \mathcal{X} \to [0,1]$ satisfying the ϕ-probabilistic-Lipschitz property with respect to the target distribution, for some function ϕ. Then, for all λ, for all $(P^S, P^T) \in \mathcal{W}$,*

$$\mathbb{E}_{S\sim P^{S^m}}[\mathrm{Err}_{P^T}(h_{\mathrm{NN}})] \leq 2\,\mathrm{opt}_{P^T} + \phi(\lambda) + \frac{4\sqrt{d}}{C\,\lambda\,m^{\frac{1}{d+1}}} \ .$$

Proof. Theorem 51 states that

$$\begin{array}{rl} & \mathbb{E}_{S\sim P^{S^m}}[\mathrm{Err}_{P^T}(h_{\mathrm{NN}})] \\ \leq & 2\,\mathrm{opt}_{P^T} + \phi(\lambda) + \lambda^{-1}\sqrt{d}\,\frac{1}{C}\left(\frac{1}{m\,\mathrm{e}\,\gamma^d} + \gamma\right) \end{array}$$

if we have a bound on the weight ratio for $\mathcal{B}_\gamma$ (see third to last line of the proof). As the bound now holds for all γ we can set $\gamma = m^{-\frac{1}{d+1}}$. This gives

$$\begin{array}{rcl} \frac{2^{d+1}\gamma^{-d}}{me} + \gamma & = & \frac{2m^{d/(d+1)}}{me} + 2m^{-1/(d+1)} \\ & = & 2m^{-1/(d+1)}(1/\mathrm{e} + 1) \leq 4m^{-1/(d+1)} \end{array}$$

and we obtain $\mathbb{E}_{S\sim P^{S^m}}[\mathrm{Err}_{P^T}(\mathrm{NN}(S))] \leq 2\,\mathrm{opt}_{P^T} + \phi(\lambda) + 4\lambda^{-1}\sqrt{d}\frac{1}{C}m^{-\frac{1}{d+1}}$.

□

Note that, if source and target data are the same, then the same analysis leads to an error bound of $\mathbb{E}_{S\sim P^m}[\mathrm{Err}_P(\mathrm{NN}(S))] \leq 2\,\mathrm{opt}_{P^T} + \phi(\lambda) + 4\lambda^{-1}\sqrt{d}m^{-\frac{1}{d+1}}$ for applying the Nearest Neighbor algorithm in the standard setting. This is the bound of Theorem 21.

For the results in this section we can actually settle for the η-weight ratio (see Definition 38). For boxes of very small target-weight, we do not need to require the source distribution to have any weight at all. More precisely, since the number of boxes we are using to cover the space in the proofs in this section is $\left(1/\gamma^d\right)$, aiming for some value of ϵ, we could waive the requirement for boxes that have target weight less than $\gamma^d\epsilon$. Thus by assuming a lower bound on the $\gamma^d\epsilon$-weight ratio, the potential misclassification of these boxes sum up to at most ϵ and thus we only produce an additional error of ϵ.

5.7 DA under Target Realizability

In many learning scenarios labeling examples can by costly and time-consuming. Thus, it is natural to investigate, whether the amount of data that is necessary for domain adaptation can be covered by unlabeled target data rather than labeled source data. We now present a domain adaptation algorithm for the case where the labeling function satisfies the (standard) Lipschitz property and the target is realizable with a margin (recall Definition 9 and Definition 10).

Note that the results of the previous section also provide a DA algorithm for labeling functions that satisfy the Lipschitz property, namely the Nearest Neighbor algorithm. However, this approach uses a large amount of labeled data (from the source distribution) and does not make use of any information from the target distribution. Here, we show that we can compensate for labeled source examples by a large unlabeled target sample if the learner has knowledge of a class that realizes the target distribution. Hereby, the size of the labeled sample required for success goes from the Nearest Neighbor sample complexity down to the (much smaller) sample complexity of standard learning under a realizability assumption.

For the results of this section, we need the notion of an ϵ-net. This was introduced by Haussler and Welzl [1986].

Definition 53 (ϵ-net, Haussler and Welzl [1986])**.** Let $\mathcal{X}$ be some domain, $\mathcal{W} \subseteq 2^{\mathcal{X}}$ a collection of subsets of $\mathcal{X}$ and P a distribution over $\mathcal{X}$. An *ϵ-net* for $\mathcal{W}$ with respect to P is a subset $N \subseteq \mathcal{X}$ that intersects every member of $\mathcal{W}$ that has P-weight at least ϵ.

Corollary 3.8 of Haussler and Welzl [1986] tells us that, there exists a constant c such that for any class $\mathcal{W}$ of finite VC-dimension, an *i.i.d.* sample from some distribution P of size at least

$$c\left(\frac{\mathrm{VCdim}(H) + \log(1/\delta)}{\epsilon} \log\left(\frac{\mathrm{VCdim}(H)}{\epsilon}\right)\right)$$

is an ϵ-net with probability at least $1 - \delta$.

We relate ϵ-nets for a source distribution to ϵ-nets for a target distribution:

Lemma 54. *Let $\mathcal{X}$ be some domain, $\mathcal{W} \subseteq 2^{\mathcal{X}}$ a collection of subsets of $\mathcal{X}$, and P^S and P^T a source and a target distribution over $\mathcal{X}$ with $C := C_{\mathcal{W}}(P^S, P^T) \geq 0$. Then every $(C\epsilon)$-net for $\mathcal{W}$ with respect to P^S is an ϵ-net for $\mathcal{W}$ with respect to P^T.*

Proof. Let $N \subseteq \mathcal{X}$ be an $(C\epsilon)$-net for $\mathcal{W}$ with respect to P^S. Consider a $U \in \mathcal{W}$ that has target-weight at least ϵ, that is $P^T(U) \geq \epsilon$. Then we have $P^S(U) \geq CP^T(U) \geq C\epsilon$. As N is an $(C\epsilon)$-net for $\mathcal{W}$ with respect to P^S, we have $N \cap U \neq \emptyset$. □

In this section, we assume the labeling function l to be deterministic.

5.7.1 Realizability with a margin

We propose the following adaptive domain adaptation procedure for the case that the learner has prior knowledge of a hypothesis class that realizes the target distribution with a margin:

Algorithm $\mathcal{A}$

Input An *i.i.d.* labeled sample S from P^S, an unlabeled *i.i.d.* sample T from $P^T_{\mathcal{X}}$ and a margin parameter γ.

Step 1 Partition the domain $[0,1]^d$ into a collection $\mathcal{B}$ of boxes (axis-aligned rectangles) with sidelength $(\gamma/\sqrt{d})$.

Step 2 Obtain sample S' by removing every point in S, which is sitting in a box that is not hit by T.

Step 3 Output an ERM classifier from H for the sample S'.

The following theorem provides upper bounds on the sizes of the labeled and the unlabeled sample that suffice for algorithm $\mathcal{A}$ to succeed. Note that the complexity of the labeled sample is comparable to the size of a labeled sample required in standard learning. It depends only on the VC-dimension, the accuracy parameters and the weight ratio between source and target.

In the statement of the theorem we use the following notation: For two collections of subsets $\mathcal{A} \subseteq 2^{\mathcal{X}}$ and $\mathcal{B} \subseteq 2^{\mathcal{X}}$ of some domain set $\mathcal{X}$, we let $\mathcal{A} \sqcap \mathcal{B}$ denote the set of all intersections of elements from $\mathcal{A}$ with elements from $\mathcal{B}$, formally

$$\mathcal{A} \sqcap \mathcal{B} := \{c \subseteq \mathcal{X} \ : \ c = b \cap a \text{ for some } a \in \mathcal{A} \text{ and } b \in \mathcal{B}\}$$

Theorem 55. *Let $\mathcal{X} = [0,1]^d$, $\gamma > 0$ a margin parameter, H be a hypothesis class of finite VC dimension and $\mathcal{W}$ be a class of pairs (P^S, P^T) of source and target distribution satisfying covariate shift with*

- *$C_{\mathcal{I}}(P^S, P^T) > 0$ for the class $\mathcal{I} = (H \Delta H) \sqcap \mathcal{B}$, where $\mathcal{B}$ is a partition of $[0,1]^d$ into boxes of sidelength $\gamma/\sqrt{d}$*
- *P^T is realizable by H with margin γ*
- *their labeling function l is a γ-margin classifier with respect to P^T.*

Then there is a constant $c > 1$ such that, for all $\epsilon > 0$, $\delta > 0$, and all $(P^S, P^T) \in \mathcal{W}$, when given an i.i.d. labeled sample S from P^S of size

$$|S| \geq c \left(\frac{\mathrm{VCdim}(H) + \log(1/\delta)}{C_{\mathcal{I}}(P^S, P^T)^2(1-\epsilon)^2\epsilon} \log \left(\frac{\mathrm{VCdim}(H)}{C_{\mathcal{I}}(P^S, P^T)^2(1-\epsilon)^2\epsilon} \right) \right)$$

and an i.i.d. unlabeled sample T from $P^T_{\mathcal{X}}$ of size

$$|T| \geq \frac{2(\sqrt{d}/\gamma)^d \ln(3(\sqrt{d}/\gamma)^d/\delta)}{\epsilon}$$

then, with probability at least $1 - \delta$, algorithm $\mathcal{A}$ outputs a classifier h with $\mathrm{Err}_{P^T}(h) \leq \epsilon$.

Proof. Let $\epsilon > 0$ and $\delta > 0$ be given and set $C = C_{\mathcal{I}}(P^S, P^T)$. We set $\epsilon' = \epsilon/2$ and $\delta' = \delta/3$ and divide the space $\mathcal{X}$ up into *heavy* and *light* boxes from $\mathcal{B}$, by defining a box $b \in \mathcal{B}$ to be light if $P^T(b) \leq \epsilon'/|\mathcal{B}| = \epsilon'/(\sqrt{d}/\gamma)^d$ and heavy otherwise. We let $\mathcal{X}^l$ denote the union of the light boxes and $\mathcal{X}^h$ the union of the heavy boxes. Further, we let P^S_h and P^T_h denote the restrictions of the source and target distributions to $\mathcal{X}^h$, that is, we have $P^S_h(U) = P^S(U)/P^S(\mathcal{X}^h)$ and $P^T_h(U) = P^T(U)/P^T(\mathcal{X}^h)$ for all $U \subseteq \mathcal{X}^h$ and $P^S_h(U) = P^T_h(U) = 0$ for all $U \nsubseteq \mathcal{X}^h$. As $|\mathcal{B}| = (\sqrt{d}/\gamma)^d$, we have $P^T(\mathcal{X}^h) \geq 1 - \epsilon'$ and thus, $P^S(\mathcal{X}^h) \geq C(1 - \epsilon')$.

We will show the following:

Claim 1 With probability at least $1 - \delta'$ an *i.i.d.* P^T-sample T of size as stated in the Theorem hits every heavy box.

Claim 2 With probability at least $1 - 2\delta'$ the intersection of S and $\mathcal{X}^h$, where S is an *i.i.d.* P^S-sample of size as stated in the theorem is an ϵ'-net for $H\Delta H$ with respect to P^T_h.

To see that these imply the claim of the theorem, let $S^h = S \cap \mathcal{X}^h$ denote the intersection of the source sample and the union of heavy boxes. By Claim 1, T hits every heavy box with high probability, thus $S^h \subseteq S'$, where S' is the intersection of S with boxes that are hit by T (see the description of the algorithm $\mathcal{A}$). Therefore, if S^h is an ϵ'-net for $H\Delta H$ with respect to P^T_h (as guaranteed by Claim 2) then so is S'. Hence, with probability at least $1-3\delta' = 1-\delta$ the set S' is an ϵ'-net for $H\Delta H$ with respect to P_h^T. Now note that an ϵ'-net for $H\Delta H$ with respect to P_h^T is an ϵ-net with respect to P^T as every set of P^T-weight at least ϵ has P_h^T weight at least ϵ', by definition of $\mathcal{X}^h$ and P_h^T.

Finally, we need to show that S' being an ϵ-net for the set $H\Delta H$ of symmetric differences with respect to the target distribution, suffices for the ERM-classifier from the class H to have target error at most ϵ. Let $h_T^* \in H$ denote the γ-margin classifier of zero target error. Note that every box in $\mathcal{B}$ of positive target weight is labeled homogeneously with label 1 or label 0 by the labeling function l as l is a γ-margin classifier as well. Let $s \in S'$ be a sample point and $b_s \in \mathcal{B}$ be the box that contains s. As h_T^* is a γ-margin classifier and $P^T(b_s) > 0$ (b_s was hit by T by the definition of S'), b_s is labeled homogeneously by h_T^* as well and as h_T^* has zero target error this label has to correspond to the labeling by l. Thus $h_T^*(s) = l(s)$ for all $s \in S'$, which means that the empirical error with respect to S' of h_T^* is zero.

Now consider a classifier h_ϵ with $\mathrm{Err}_{P^T}(h_\epsilon) \geq \epsilon$. Let $s \in S'$ be a sample point in $h_T^*\Delta h_\epsilon$ (which exists as S' is an ϵ-net). As $s \in h_T^*\Delta h_\epsilon$, we have $h_\epsilon(s) \neq h_T^*(s) = l(s)$ and thus, h_ϵ as an empirical error larger than zero, which implies that no classifier of error larger than ϵ can be chosen by ERM on input S'.

Proof of Claim 1: Let b be a heavy box, thus $P^T(b) \geq \epsilon'/|\mathcal{B}|$. Then, when drawing an *i.i.d.* sample T from P^T, the probability of not hitting b is at most $(1-(\epsilon'/|\mathcal{B}|))^{|T|}$. Now the union bound implies that the probability that there is at a box in $\mathcal{B}^h$ that does not get hit by the sample T is bounded by

$$|\mathcal{B}^h|(1-(\epsilon'/|\mathcal{B}|))^{|T|} \leq \ |\mathcal{B}|(1-(\epsilon'/|\mathcal{B}|))^{|T|} \leq |\mathcal{B}|\mathrm{e}^{-\epsilon'|T|/|\mathcal{B}|}.$$

Thus if

$$|T| \geq \frac{|\mathcal{B}|\ln(|\mathcal{B}|/\delta')}{\epsilon'} = \frac{2(\sqrt{d}/\gamma)^d \ln(3(\sqrt{d}/\gamma)^d/\delta)}{\epsilon}$$

the sample T hits every heavy box with probability at least $1-\delta'$.

Proof of Claim 2: Let $S^h := S \cap \mathcal{X}^h$. Note that, as S is an *i.i.d.* P^S sample, we can consider S^h to be an *i.i.d.* P_h^S sample. We have the following bound on the weight ratio between P_h^S and P_h^T:

$$\begin{aligned} C_{\mathcal{I}}(P_h^S, P_h^T) = &\ \inf_{p\in\mathcal{I}, P_h^T(p)>0} \tfrac{P_h^S(p)}{P_h^T(p)} = \inf_{p\in\mathcal{I}, P_h^T(p)>0} \tfrac{P^S(p)}{P^T(p)} \tfrac{P^T(\mathcal{X}^h)}{P^S(\mathcal{X}^h)} \\ \geq &\ C \tfrac{P^T(\mathcal{X}^h)}{P^S(\mathcal{X}^h)} \geq C(1-\epsilon'), \end{aligned}$$

where the last inequality holds as $P^T(\mathcal{X}^h) \geq (1-\epsilon')$ and $P^S(\mathcal{X}^h) \leq 1$. Note that every element in $H\Delta H$ can be partitioned in to elements from $\mathcal{I}$, therefore we obtain the same bound on the weight ratio for the symmetric differences of H:

$$C_{H\Delta H}(P_h^S, P_h^T) \geq C(1-\epsilon').$$

It is well known that there is a constant $c > 1$ such that, conditioned on S^h having size at least

$$M := c\left(\frac{\mathrm{VCdim}(H\Delta H) + \log(1/\delta')}{C(1-\epsilon')\epsilon'} \log\left(\frac{\mathrm{VCdim}(H\Delta H)}{C(1-\epsilon')\epsilon'}\right)\right),$$

with probability at least $1-\delta'$ it is a $C(1-\epsilon')\epsilon'$-net with respect to P_h^S and thus an ϵ'-net with respect to P_h^T by Lemma 54 (see Definition 53 and discussion above).

Thus, it remains to show that with probability at least $1-\delta'$ we have $|S^h| \geq M$. As we have $P^S(\mathcal{X}^h) \geq C(1-\epsilon')$, we can view the sampling of the points of S and checking whether they hit $\mathcal{X}^h$ as a Bernoulli variable with mean $\mu = P^S(\mathcal{X}^h) \geq C(1-\epsilon')$. Thus, by Hoeffding's inequality we have that for all $t > 0$

$$\Pr(\mu|S| - |S^h| \geq t|S|) \leq \mathrm{e}^{-2t^2|S|}.$$

If we set $C' = C(1-\epsilon')$, assume $|S| \geq \frac{2M}{C'}$ and set $t = C'/2$, we obtain

$$\Pr(|S^h| < M) \ \leq \ \Pr(\mu|S| - |S^h| \geq \frac{C'}{2}|S|) \ \leq \ \mathrm{e}^{-\frac{C'^2|S|}{2}}.$$

Now

$$|S| \geq \frac{2M}{C'} > \frac{2(\mathrm{VCdim}(H\Delta H) + \log(1/\delta'))}{C^2(1-\epsilon')^2\epsilon'}$$

implies that $\mathrm{e}^{-\frac{C'^2|S|}{2}} \leq \delta'$, thus we have shown that S^h is an ϵ'-net of $H\Delta H$ with probability at least $(1-\delta')^2 \geq 1-2\delta'$.

Imitating the proof of Claim 1 by Ben-David and Litman [1998] one can show that $\mathrm{VCdim}(H\Delta H) \leq 2\mathrm{VCdim}(H) + 1$. This completes the proof. □

5.7.2 Finite domain

The procedure $\mathcal{A}$ from the previous section can be modified to work on any finite domain with arbitrary labeling functions and hypothesis classes of finite VC-dimension (under the target-realizability assumption). For the modification, we delete Step 2 and instead of Step 3 the algorithm removes every point from the labeled source sample S which is not hit by the unlabeled target sample T. This does not change the size of the source sample S needed for a guarantee of success, but the size of the target sample now depends on the size of the domain instead of the labeling function's Lipschitzness. The proof of the following result is a simple modification of the proof of Theorem 55 and is left to the reader.

Theorem 56. *Let $\mathcal{X}$ be some domain, H be a hypothesis class of finite VC dimension and $\mathcal{W} = \{(P^S, P^T) \mid C(P^S_{\mathcal{X}}, P^T_{\mathcal{X}}) > 0, \mathrm{opt}_{P_T}(H) = 0\}$ be a class of pairs of source and target distributions satisfying covariate shift, with bounded weight ratio and the target being realizable by H. Then there is a constant $c > 1$ such that, for all $\epsilon > 0$, $\delta > 0$, and all $(P^S, P^T) \in \mathcal{W}$, when given an i.i.d. labeled sample S from P^S of size*

$$|S| \geq c\left(\frac{\mathrm{VCdim}(H) + \log(1/\delta)}{C(P^S, P^T)^2(1-\epsilon)^2\epsilon}\log\left(\frac{\mathrm{VCdim}(H)}{C(P^S, P^T)^2(1-\epsilon)^2\epsilon}\right)\right)$$

and an i.i.d. unlabeled sample T from $P^T_{\mathcal{X}}$ of size

$$|T| \geq \frac{2|\mathcal{X}|\ln(3|\mathcal{X}|/\delta)}{\epsilon}$$

then algorithm $\mathcal{A}$ outputs a classifier h with $\mathrm{Err}_{P^T}(h) \leq \epsilon$ with probability at least $1 - \delta$.

5.8 Proper DA Learning

In this section, we show that in the context of proper domain adaptation, the use of algorithms that utilize target generated data, is necessary. We show that there are classes that can not be properly learned without access to data from the test distribution:

Theorem 57. *Let our domain set be the unit cube in $\mathbb{R}^d$, for some d. Consider the class H of half-spaces as our target class. Let x and z be a pair of antipodal points on the unit* *sphere and let $\mathcal{W}$ be a set that contains two pairs (P^S, P^T) and $(P^S, P^{T'})$ of distributions with:*

1. *both pairs satisfy the covariate shift assumption,*
2. *$l(x) = l(z) = 1$ and $l(\bar{0}) = 0$ for their common labeling function l,*
3. *$P^S_{\mathcal{X}}(x) = P^S_{\mathcal{X}}(z) = P^S_{\mathcal{X}}(\bar{0}) = 1/3$,*
4. *$P^T_{\mathcal{X}}(x) = P^T_{\mathcal{X}}(\bar{0}) = 1/2$ and $P^{T'}_{\mathcal{X}}(z) = P^{T'}_{\mathcal{X}}(\bar{0}) = 1/2$.*

Then, for any $\epsilon < 1/2$, any $\delta < 1/2$ and any number m, no DA learning algorithm can $(\epsilon, \delta, m, 0)$ solve the domain adaptation task of properly learning H with respect to $\mathcal{W}$.

Proof. Clearly, no halfspace can correctly classify the three points, $x, \bar{0}$ and y. Note that for any halfspace h, we have $\mathrm{Err}_{P^T}(h) \geq 1/2$ or $\mathrm{Err}_{P^{T'}}(h) \geq 1/2$ (or both). Thus for every DA learner, there exists a target distribution (either P^T or $P^{T'}$) such that, with probability at least $1/2$ over the sample, outputs a function of error at least $1/2$. Lastly, note that the approximation error of the class of halfspaces for the target distributions is 0. □

This theorem implies that no conservative DA learner succeeds in domain adaptation learning of halfspaces even with respect to the simple class of distributions $\mathcal{W}$. Note that both target distributions in $\mathcal{W}$ are realizable by the class of halfspaces. Thus, we obtain the following result (recall that we call a DA-learner *conservative* if it does not make use target generated data, see Section 5.3):

Corollary 58. *No conservative DA-learner can properly learn the class of halfspaces over the domain $\mathcal{X} = [0,1]^d$ with respect to a class $\mathcal{W}$ of pairs of source and target distributions that satisfy covariate shift, have weight ratio larger than C for some $0 < C < 2/3$ and where the target is realizable by H.*

In the example of Theorem 57 it becomes crucial for the learning algorithm to estimate whether the support of the target distribution is on x and $\bar{0}$ or on z and $\bar{0}$. This information cannot be obtained without access to a sample of the target distribution despite of a pointwise weight ratio as large as $1/2$. Thus, no amount of source generated data can compensate for having a sample from the target distribution.

We now present a general method for proper DA learning. The idea of our construction is to apply a simple two step procedure, similar to the one suggested Chapter 4 in the context of semi-supervised learning. In the first step, we use the labeled examples from the source distribution to learn an arbitrary predictor, which should be rather accurate on the target distribution. For example, as we have shown in the previous section, this predictor can be the Nearest Neighbor rule. In the second step, we will apply that predictor to the unlabeled examples from the target distribution and feed this constructed (now labeled) sample to a standard agnostic learner for the usual supervised learning setting.

As in our results for proper semi-supervised learning we make use of Lemma 26, which quantities the robustness of agnostic learners to small changes in the labeling function of the data generating distribution. This lemma implies that, in order to prove the success of our DA algorithm, it suffices to show that the classifier that we learn in the first step has error smaller than $\epsilon/3$ with confidence at least $1 - \delta/2$. If we then use an agnostic learner for our output-class H in the second step and feed this agnostic learner with a sample of the size it needs to $(\epsilon/3, \delta/2)$-learn H, our 2-step procedure (ϵ, δ)-learns H. Applying this, we readily get:

Theorem 59. *Let $\mathcal{X}$ be some domain and $\mathcal{W}$ be a class of pairs (P^S, P^T) of distributions over $\mathcal{X} \times \{0, 1\}$ with $\mathrm{opt}_{P^T} = 0$ such that there is an algorithm $\mathcal{A}$ and functions $m : (0, 1)^2 \to \mathbb{N}$, $n : (0, 1)^2 \to \mathbb{N}$ such that $\mathcal{A}$ $(0, \epsilon, \delta, m(\epsilon, \delta), n(\epsilon, \delta))$-solves the domain adaptation learning task for $\mathcal{W}$ for all $\epsilon, \delta > 0$. Let H be some hypotheses class for which there exists an agnostic proper learner. Then, the H-proper domain adaptation problem with respect to the class $\mathcal{W}$ can be $(1, \epsilon, \delta, m(\epsilon/3, \delta/2), n(\epsilon/3, \delta/2) + m'(\epsilon/3, \delta/2))$-solved, where m' is the sample complexity function for agnostically learning H.*

Proof. Given the parameters ϵ and δ, let S be a P^S-sample of size at least $m(\epsilon/3, \delta/2)$ and T be an unlabeled $P^T_{\mathcal{X}}$-sample of size $n(\epsilon/3, \delta/2) + m'(\epsilon/3, \delta/2)$. Divide the unlabeled sample into a sample T_1 of size $n(\epsilon/3, \delta/2)$ and T_2 of size $m'(\epsilon/3, \delta/2)$. Apply $\mathcal{A}(S, T_1)$, the predictor resulting form applying the learner $\mathcal{A}$ to the S and T_1, to label all members of

T_2, and then feed the now-labeled T_2 as input to the agnostic proper learner for H. The claimed performance of the output hypothesis now follows from Lemma 26. □

The algorithm $\mathcal{A}$ used in this theorem could be the Nearest Neighbor algorithm, if the class $\mathcal{W}$ satisfies the conditions for Theorem 51. In summary, we have shown that with a non-conservative DA algorithm, that employs unlabeled examples from the target distribution, we can agnostically learn a member of the hypotheses class for the target distribution, whereas without target generated data we can not.

5.9 Discussion

We have analyzed domain adaptation learning in the setting where the learner has access to labeled source and unlabeled target data. Our lower bounds in Section 5.5 imply that DA learning is impossible unless at least one of these types of samples is available in very large quantities. This holds even for DA learning under assumptions that are so strong that they may seem to trivialize the problem. Thus any positive learning guarantees for DA with reasonable samples sizes will hold only under rather strong assumptions on the relatedness between source and target task.

The type of data that is easier to obtain for a DA learning task, determines whether it is more appropriate to employ a conservative DA algorithm (that uses only data from the source distribution) or to choose an adaptive paradigm. We have presented algorithms (and discussed previous work on) both these types of approaches.

The Nearest Neighbor algorithm discussed in Section 5.6 is a conservative approach to DA. It labels target (test-)points solely by their nearest neighbor in a source generated sample. We have shown that this is successful if source and target distribution have bounded weight ratio with respect to the set of axis-alligned rectangles, and if the size of the source-generated sample is large enough to compensate for the discrepancy between source and target (as quantified by the weight ratio). Thus, here DA is possible with a (rather large) source sample without access to target data.

We have further shown that, if labeling is expensive but there is unlabeled data from the target data, then a large unlabeled target sample can reduce the necessary amount of labeled data back to what is also needed in the standard setting (its size depending on the VC-dimension of the class to be learned and the weight ratio between source and target). In Section 5.7 we present a reweighting paradigm that achieves this under the assumption that the target task is realizable with a margin by some hypothesis class and source and target have bounded weight ratio with respect to the set of symmetric differences of the class. Thus, for this setting, DA is possible with a relatively small number of labels provided the unlabeled target data. Note that our lower bound in Theorem 42 applies to this setting and therefore DA is not possible with this small amount of labeled data without the large unlabeled sample.

Ben-David et al. [2012] also show that, if the learner has prior knowledge that the source distribution is realizable by some class and source and target have bounded weight ratio with respect to the set of symmetric differences in this class, then DA is also possible by simply applying a classifier that is learned from source data to the target (the error is worsened by a factor that corresponds to the weight ratio in this case). More precisely:

Observation 60 (Ben-David et al. [2012])**.** *Let $\mathcal{X}$ be a domain and let P^S and P^T be a source and a target distribution over $\mathcal{X} \times \{0,1\}$ satisfying the covariate shift assumption, with $C_{H\Delta H}(P^S_{\mathcal{X}}, P^T_{\mathcal{X}}) > 0$. Then we have $\mathrm{Err}_{P^T}(h) \leq \frac{1}{C_{H\Delta H}(P^S_{\mathcal{X}}, P^T_{\mathcal{X}})} \mathrm{Err}_{P^S}(h)$ for all $h : \mathcal{X} \to \{0,1\}$.*

Combining the lower and upper bounds described above, we conclude that when the size of the source generated labeled sample is supposed to be small (i.e. independent of the domain size) then, without strong prior knowledge about the source distribution, successful DA is possible only by adaptive algorithms (algorithms that make use of target generated data). In other words, in that setup, unlabeled target generated data is provably necessary and beneficial for DA learning.

It is interesting to note that in the setting of proper learning, DA is impossible without access to target generated DA independently of the amount of source generated (labeled) data. We formally proved this in Section 5.8.

Chapter 6

Active Learning

6.1 Overview

In Active Learning (AL), the learner gets a sample of unlabeled instances as input. It can then successively choose points from that sample and obtains their labels. The learner's choice of points for label query is based on the instances in the unlabeled input sample and, at each step, on the previously seen labels. By choosing the instances wisely, the learner aims to achieve good prediction performance while using as little label queries as possible, and, in particular, less than the amount of labeled examples required for learning from a fully labeled random sample.

Active learning paradigms are successful in practice and there is also a variety of theoretical studies analyzing the possibilities and limitations of AL. However, several studies have shown that, under worst-case scenarios, AL algorithms are bound to require as many labeled sample points as their "passive" fully supervised counterparts (see discussion of related work in in Section 6.2). Those negative results set the frame for most of the theoretical research on AL. Rather than trying to show that active choice of label queries can *always* reduce the number of training labels, one aims to identify properties of the learning task under which an AL paradigm is beneficial. In this chapter, we show that allowing active label queries provably reduces the labeled sample complexity of learning tasks, if the data generating distribution satisfies Probabilistic Lipschitzness.

We present a procedure that takes an unlabeled sample, queries the (true) labels of some of the sample points, and returns a full (but possibly erroneous) labeling of the input sample. This algorithm (PLAL) follows a paradigm proposed by Dasgupta and Hsu [2008] for exploiting cluster structure in active learning. While most previous work on the theory of active learning focused on an efficient version space reduction for learning a hypothesis class, Dasgupta and Hsu [2008] suggest a labeling procedure based on a hierarchical clustering of the training data. The authors of that study show that, assuming that the learner is given a "good" hierarchical clustering, an unlabeled sample can be labeled almost correctly with relatively few label queries. They suggest to then feed the now labeled sample to any standard learning procedure. Here, we analyze a version of their approach under the assumption of PL. This condition is weaker than the availability of a "good" clustering tree in that we only need it for the analysis of our procedure (as opposed to the need for a successful preprocessing step that finds the good clustering tree).

Summary of our results

Our main result is proving reductions in labeled sample complexity for active learning with PLAL in comparison to standard (passive) learning for several learning paradigms.

Our labeling procedure, PLAL, is presented in Section 6.4. We show here that PLAL assigns incorrect labels to at most an ϵ-fraction of the original unlabeled sample, where ϵ is an error parameter provided as input to the algorithm. Furthermore, we show that, under the assumption of bounded PL, PLAL queries labels only for a fraction of the input sample points. We derive bounds on the number of label queries in terms of the input error parameter and the size of the input unlabeled sample.

In Section 6.5, we analyze under which conditions PLAL can be used as a preprocedure to other learning paradigms to reduce the their label complexity. We argue in his section that it is *safe* (in that it does not increase the error of the learned classifier by too much) to use PLAL as a pre-procedure to a variety of learning methods: When learning a hypothesis class, it is safe to use PLAL prior to any ERM or RLM learner. Further, we show that it is safe to use PLAL for Nearest Neighbor learning.

The labeling-error and query number bounds for PLAL together with proving that using

PLAL can not increase the error of the learned classifier by much, result in upper bounds for the labeled sample complexity of PLAL-based active learning under PL. Combining these with the lower bounds for passive learning under Probabilistic Lipschitzness from Chapter 3, we show that, under polynomial rates of Probabilistic Lipschitzness, PLAL significantly reduces the labeled sample complexity of learning some VC-classes as well as unrestricted (Nearest Neighbor based) learning.

We start with discussing previous related work in Section 6.2. In Section 6.3 we present our formal framework for active learning. The content of this chapter was published by Urner et al. [2013].

6.2 Related Work

Active learning is a very lively area of research. There are many, practically relevant techniques for employing active queries in order to reduce the number of labels needed. Settles [2009] provides an overview and comparison of available methods. We focus on discussing previous work on the analysis of active learning in the framework of statistical learning theory.

The theoretical analysis of active learning was motivated by a simple example of learning initial segments on the line, where binary search leads to an exponential improvement of labeled sample complexity from $1/\epsilon$ to $\log(1/\epsilon)$ in the realizable case, see Dasgupta [2004]. This has inspired a large body of work, that analyses how active queries can reduce the number of labels needed to reduce a hypothesis space to only close to optimal functions, by eliminating all functions that have non-zero empirical error. Dasgupta [2005] showed that this is not possible with less than $1/\epsilon$ queries in general. Depending on the target concept the labeled sample complexity of a specific task (learning linear separators under uniform marginal) can vary between $O(\log(1/\epsilon))$ and $\Omega(1/\epsilon)$. Dasgupta [2004] and Dasgupta et al. [2003] show that a greedy label querying strategy for finding a hypothesis that is consistent with a sample approximates the optimal strategy up to a factor of $\log(|H|)$. Adler and Heeringa [2012] that the optimal strategy can not be approximated to arbitrarily small factors.

Research has thus mostly focused on finding properties and parameters of learning tasks (distributions, hypothesis classes and combinations of these) that allow for provable reductions in sample complexity by active learning methods. Dasgupta [2005] introduced the *splitting index*, a parameter that relates a hypothesis class to a marginal distribution, and presents labeled sample complexity bounds for active learning in the realizable case in terms of this parameter. Balcan et al. [2007] investigate active learning of linear separators under a margin assumption and uniform marginal distribution. For the case of learning linear separators under a margin assumption the greedy strategy has recently been revisited by Gonen et al. [2013].

There have been extensive efforts to generalize the positive results for active learning from the realizable to the agnostic case. Lower bounds of $\Omega(1/\epsilon^2)$ by Kääriäinen [2006] imply that, again, improvements in labeled sample complexity for learning a hypothesis class are not possible in general. Thus, here as well, research focuses on identifying parameters that characterize learning tasks where active learning is beneficial. The, so far, most prominent such parameter is *disagreement coefficient*, introduced by Hanneke [2007]. Wang [2011] presents some upper and lower bound on the disagreement coefficient for classes with smooth classification boundaries with respect to distributions that have smooth densities. The disagreement coefficient has been used to bound the labeled sample complexity of various querying strategies (Hanneke [2007], Dasgupta et al. [2008], Beygelzimer et al. [2010], Beygelzimer et al. [2009]). However, the bounds on the number of label queries in these papers all involve the approximation error of the hypothesis class. They become relevant only when the approximation error is small, the learning task thus close to the realizable case.

The survey "The two faces of active learning" by Dasgupta [2011] contrasts two general approaches for active learning: Using label queries to more efficiently search through a hypothesis space and exploiting cluster structure in data. Almost all of the theoretical work so far has focused on the former setup. A first approach at exploiting cluster structure by active learning has been presented by Dasgupta and Hsu [2008]. This study presents a labeling strategy for an unlabeled data, when the learner is also given a hierarchical clustering of the data. It is shown that the suggested strategy mislabels at most an ϵ-fraction of the points. The authors suggest that the labeling of the sample points, that is hereby obtained, could then be used for a "second round of learning", however no performance guarantees for this second round of learning are given. Furthermore, the bound on the number of labels depends on the depth of the effectively used clustering tree and it is unclear how to control this parameter. Our work builds on the ideas of Dasgupta and Hsu [2008]. We provide a clean analysis of the labeling strategy and show how to use it for the second round of learning in a way that admits provable performance guarantees and reduction of labeled sample complexity under the assumption of Probabilistic Lipschitzness. We bound the number of queries of our algorithm in terms of the Probabilistic Lipschitzness of the underlying data distribution (independently of the depth of the resulting cluster tree and independently of the approximation error of a class).

A framework, where an unlabeled sample is labeled by a preliminary, active labeling procedure and then fed to a standard learner has been introduced by Hanneke [2012] under the title "activized learning". Assuming the data is realizable by a VC-class, the author presents labeling procedures based on repeated computations of the shatter function of version spaces and shows how this labeling procedure reduces the labeled sample complexity of the original standard learner. Our PLAL procedure achieves labeled sample complexity reduction results for data with bounded PL, which we believe is a more realistic assumption, and is substantially simpler and easy to implement.

6.3 Formal Framework for Active Learning

In this chapter, we consider domain sets $\mathcal{X} = [0,1]^d$ for some dimension d, and distributions P over $\mathcal{X} \times \{0,1\}$ with a deterministic labeling function $l : \mathcal{X} \to \{0,1\}$.

An *active learner* receives an unlabeled sample $S_{\mathcal{X}} = (x_1, \dots, x_m)$ generated *i.i.d.* by $P_{\mathcal{X}}$ (throughout this chapter, we use the notation $S_{\mathcal{X}}$ as opposed to S, if we want to stress that a sample consists of *unlabeled* domain points). The active learner can then sequentially query labels for points in $S_{\mathcal{X}}$, that is, the learner chooses indices $i_1, \dots, i_n \in \{1, \dots m\}$ and receives the labels $l(x_{i_1}), \dots, l(x_{i_n})$. At each step, the choice of each i_j can depend on $S_{\mathcal{X}}$ and the labels seen so far. Based on the unlabeled sample $S_{\mathcal{X}}$ and the queried labels, the learner outputs a hypothesis.

We say that an algorithm $\mathcal{A}$ *actively learns* some hypothesis class H over $\mathcal{X}$ with respect to a set of distribution $\mathcal{Q}$ over $\mathcal{X} \times \{0,1\}$, if there exist functions $m_u : (0,1) \times (0,1) \to \mathbb{N}$, $m_l : (0,1) \times (0,1) \to \mathbb{N}$, such that, for all $\epsilon, \delta \in (0,1)$, for all distributions $P \in \mathcal{Q}$, with probability at least $1-\delta$ over an *i.i.d.* unlabeled $P_{\mathcal{X}}$-generated sample $S_{\mathcal{X}}$ of size $m_u(\epsilon,\delta)$, the algorithm $\mathcal{A}$ queries at most $m_l(\epsilon,\delta)$ members of S for their labels and $\mathrm{Err}_P(\mathcal{A}(S_{\mathcal{X}})) \leq \mathrm{Err}_P(H) + \epsilon$. Given a function m_u for the size of the unlabeled sample, we say that $\mathcal{A}$ has *labeled sample complexity* or *label complexity* m_l *with respect to* m_u for the smallest function m_l such that the pair of functions (m_l, m_u) that satisfies the above condition. The minimum such function m_l for which there exists a function m_u such that the pair of functions (m_l, m_u) satisfies the above condition is called the *labeled sample complexity* of $\mathcal{A}$ *for actively learning* H with respect to $\mathcal{Q}$ and denoted by $m^{\mathrm{act}}[\mathcal{A}, H, \mathcal{Q}]$. We define $m^{\mathrm{act}}[H, \mathcal{Q}]$ and $m^{\mathrm{act}}[H, \mathcal{Q}, \mathrm{proper}]$ analogously to the passive counterparts in Section 2.2.2.

6.4 The PLAL Labeling Procedure

The framework for our algorithm was suggested by Dasgupta and Hsu [2008]. The idea is to use a hierarchical clustering (cluster tree) of the unlabeled data, check the clusters for label homogeneity by starting at the root of the tree (the whole data-set) and working towards the leaves (single data points). The label homogeneity of a cluster is estimated by choosing data points for label queries uniformly at random from the cluster. If a cluster can be considered label homogeneous with sufficiently high confidence, all remaining unlabeled points in the cluster are labeled with the majority label and no further points from this cluster will be queried. If a cluster is detected to be label heterogeneous, it is split into its children in the pre-defined cluster tree. Since the cluster tree is fixed before any labels were seen, the algorithm can reuse labels from the parent cluster (the induced subsample can be considered a sample that was chosen uniformly at random from the points in the child-cluster) without introducing any sampling bias. Dasgupta [2011] provides a nice overview on this.

Dasgupta and Hsu [2008] analyze this framework and provide label-query bounds assuming that there exist a label homogeneous clustering of the data consisting of a relatively small number of tree-node clusters. In contrast, our analysis depends on the rate in which the diameters of the clusters shrink. Invoking the PL assumption, we can turn such cluster-diameter bounds into error bounds and label query bounds of the procedure. The rates in which cluster diameters shrink have been analyzed for cluster trees that are induced by *spatial trees* by Verma et al. [2009]. In our work, we consider a version of the general framework that employs spatial trees for the hierarchical clustering. To obtain a concrete algorithm from the general framework, we also need to specify how many points to query per cluster, and in which order to choose the clusters. We describe our version of this labeling procedure in the next subsection.

6.4.1 The algorithm

A spatial tree is a binary tree $\mathcal{T}$, where each node consists of a subset of the space $\mathcal{X} = [0,1]^d$. We refer to these subsets as *cells*. The root $\mathrm{Root}(\mathcal{T})$ of a spatial tree is the whole space $[0,1]^d$ and for each node (cell) C the children $\mathrm{Left}(C)$ and $\mathrm{Right}(C)$ form a 2-partition of the node C. This implies that for each *level* (distance from the root) k, the nodes at this level form a 2^k-partition of the space. For a sample S, a spatial tree induces a hierarchical clustering of S with *clusters* $S \cap C$ for the nodes C in the tree.

Our algorithm works in rounds (see pseudocode in Algorithm 1). It takes an unlabeled *i.i.d.* sample $S_{\mathcal{X}}$ and a spatial tree $\mathcal{T}$ as input. At each round, the algorithm maintains a partition of the space $[0,1]^d$ into *active* and *inactive cells*. Initially, there is only one active cell, which is the root of the tree $\mathcal{T}$, that is, the entire unit cube $[0,1]^d$ containing all sample points. Per round (level), the algorithm queries sufficiently many labels from the $S_{\mathcal{X}}$ points in each of the active cells, to detect if the cell is label heterogeneous (the next paragraph gives a more detailed explanation for this method, called $C.query()$ in line 7 of the pseudocode). A *label homogenous* cell (all seen labels in the cell are the same) is declared inactive and all remaining sample points in the cell are assigned that label. For a *label heterogeneous* cell, the children of the cell in $\mathcal{T}$ are added to the list of active cells for the next round, if they still contain unlabeled points.

For a cell C at level k, method $C.query(q_k)$ queries the labels of the first q_k sample points in the cell. For this, it reuses labels of points that were queried in earlier rounds

Algorithm 1 PLAL labeling procedure

```
Input: unlabeled sample $S_{\mathcal{X}} = (x_1, \ldots, x_m)$, spatial tree $\mathcal{T}$, parameters $\epsilon, \delta$
level = 0
$active\_cells[0].append(\mathrm{Root}(\mathcal{T}))$
while $active\_cells[\mathrm{level}]$ not empty do
    $q_{\mathrm{level}} = \frac{\mathrm{level} \cdot 2 \cdot \ln(2) + \ln(1/\delta)}{\epsilon}$
    for all $C$ in $active\_cells[\mathrm{level}]$ do
        $C.query(q_{\mathrm{level}})$
        if all labels seen in $C$ are the same then
            label all points in $C \cap S$ with that label (cell $C$ now becomes inactive)
        else
            if there are unqueried points in $C \cap S$ then
                $active\_cells[\mathrm{level} + 1].append(\mathrm{Right}(C), \mathrm{Left}(C))$
            end if
        end if
    end for
    level = level + 1
end while
Return: labeled sample $S = ((x_1, y_1) \ldots, (x_m, y_m))$
```

(that is, does not actually query those). We call the sequence $(q_k)_{k \in \mathbb{N}}$ the *query numbers*. If the cell contains fewer than q_k sample points, the labels of all unlabeled points among these are queried and the cell is declared inactive. In this case, it is not important whether the cell is label homogeneous or label heterogeneous, as the algorithm does not infer labels for any of the points and thus all the labels of points in such cells are correct labels. Note that "declaring a cell inactive" is implicit in the code of Algorithm 1: Only for cells that are heterogeneous *and* contain unlabeled points the children are added to the list of active cells for the next round.

At the end of the procedure all sample points in $S_{\mathcal{X}}$ are labeled. Each point was either queried or obtained an induced label from the homogeneous declared cell it resides in. Only in the latter case, a point might possibly have obtained an erroneous label. We show in

Subsection 6.4.2 below that, by setting the query numbers to $q_k = \frac{k \cdot 2 \cdot \ln(2) + \ln(1/\delta)}{\epsilon}$, we can bound the number of labeling mistakes this algorithm makes.

6.4.2 Error bound

In this section, we prove that with high probability over the unlabeled input sample, PLAL will label almost all points in the sample correctly. This error guarantee holds without any assumptions on the data-generating process (that is, in particular, independently of Probabilistic Lipschitzness). We use the following simple probability calculation in order to ensure, that PLAL will not oversee label-heterogeneity of a sub-sample in a cell:

Lemma 61. *Let $\mathcal{X}$ be a domain, $P_{\mathcal{X}}$ a distribution over $\mathcal{X}$, and $l : \mathcal{X} \to \{0, 1\}$ a labeling function. For any $\epsilon, \delta > 0$, if l assigns both labels with $P_{\mathcal{X}}$-probability larger than ϵ, then with probability at least $1 - \delta$ in an i.i.d. $P_{\mathcal{X}}$-sample, labeled by l, of size at least*

$$\frac{\ln(2/\delta)}{\epsilon}$$

both labels will occur.

Proof. By our assumption, each label is assigned with probability at most $(1-\epsilon)$ according to $P_{\mathcal{X}}$. Thus, for each label, the probability that we do not see a point of this (fixed) label in a sample of size m is bounded by

$$(1 - \epsilon)^m \leq \mathrm{e}^{-\epsilon m}.$$

Thus, the probability that we either do not see a 1 or do not see a 0 is bounded by

$$2\mathrm{e}^{-\epsilon m}.$$

If $m \geq \frac{\ln 2/\delta}{\epsilon}$ then $2\mathrm{e}^{-\epsilon m} \leq \delta$. Thus, the probability that we do not see both labels although both labels have weight at least ϵ is less than δ. □

For arguing that PLAL will detect label heterogeneity in the subsample S_C of a cell C, we would like to employ the statement in the above lemma with the uniform distribution over this subsample S_C. Then, sampling from this distribution, corresponds to choosing points from S_C uniformly at random with replacement. However, we suggest to query the labels of the first q_k (where k is the level of cell C) sample points in S_C according to the order of points in the original sample. This corresponds to sampling from S_C without replacement. Note that sampling without replacement will only increase the probability of seeing both labels, since the fraction of a label gets smaller while we are only seeing this one label (that is, in the above proof, the probability seeing one (fixed) label is still bounded by $(1-\epsilon)^m$). This type of sampling further allows us to consider the set of points from S_C whose labels were queried as as a sample from the data generating distribution restricted to the cell C, which in turn means that we are also testing the label-heterogeneity of the underlying distribution on this cell (also see Remark 63 below).

We can now present the following bound on the labeling errors of PLAL:

Theorem 62. *Let $\mathcal{X} = [0,1]^d$ be the domain, $P_\mathcal{X}$ a distribution over $\mathcal{X}$, $l : \mathcal{X} \to \{0,1\}$ a labeling function and $m \in \mathbb{N}$. Then, when given an i.i.d. unlabeled $P_\mathcal{X}$-sample $S_\mathcal{X}$ of size m and parameters ϵ and δ, with probability at least $(1-\delta)$ (over the choice of the sample $S_\mathcal{X}$), PLAL labels at least $(1-\epsilon)m$ many points from $S_\mathcal{X}$ correctly.*

Proof. Consider a cell that is declared inactive by the PLAL procedure. This cell was either declared homogeneous or all the points in the cell were actually queried for their label. In the latter case, all points receive the correct label. We show that in each cell C, that was declared homogeneous, at most an ϵ-fraction of the points are labeled incorrectly.

Lemma 61 together with the ensuing discussion shows that, for any cell C, if $\min\{\Pr[l = 1|C], \Pr[l = 0|C]\} \geq \epsilon$, then PLAL choosing a subsample of size $\frac{\ln(2/\delta)}{\epsilon}$ for label query has probability at most δ of seeing only one label. Therefore, choosing query numbers $\frac{\ln(2/\delta_C)}{\epsilon}$, for every cell C, guarantees that with probability at least $1-\delta_C$, it will either be declared homogeneous, resulting in at most an ϵ-fraction of the sample points in the cell being misclassified or the cell will be declared heterogeneous and split further. By choosing $\delta_C = \delta/2^{2k-1}$, where k is the level of the cell C, we ensure that the sum over all confidence parameters δ_C for all cells C, that are declared homogeneous, is at most δ (note this results in our query numbers $\frac{\ln(2/\delta_C)}{\epsilon} = \frac{k\cdot 2\cdot\ln(2)+\ln(1/\delta)}{\epsilon}$) Thereby, with probability $1-\delta$ over samples, PLAL labels at least a $(1-\epsilon)$-fraction of the points correctly. □

Remark 63. It is interesting to note that, if the spatial tree was fixed *before* the unlabeled sample $S_{\mathcal{X}}$ was drawn, then for a given cell C the set of points whose labels were queried can be viewed as a sample from the underlying distribution restricted to this cell. This implies that, when PLAL declares the sample in a cell label homogeneous (after querying the labels of the first $\frac{\ln(2/\delta_C)}{\epsilon}$ sample points in the cell), we can actually conclude that at most an ϵ-fraction (according to the distribution) of all domain points in the cell are of the opposite label. Thus, if we restrict our view to the cells that get declared homogeneous during a run of PLAL, the labeling that labels those cells with the detected label has error at most ϵ (with high probability over the sample). We use this, when arguing that we can use PLAL as a preprocedure to Nearest Neighbor learning in Section 6.5.2.

6.4.3 Bound on the number of queries

We now provide a bound on the number of queries the algorithm makes when fed with an unlabeled sample of size m under the assumption that the data generating distribution satisfies a Probabilistic Lipschitz condition. Our bounds involve the spread of the sample points at level k, called the *data diameter*. In order to avoid overloaded notation, we consider the spatial tree $\mathcal{T}$ fixed for this definition. For a set of unlabeled points $S_{\mathcal{X}}$, we let $\lambda_k^{S_{\mathcal{X}}}$ denote the maximum data-diameter in a cell at level k, that is, $\lambda_k^{S_{\mathcal{X}}} = \max\{\mathrm{diam}(C, S_{\mathcal{X}}) \,:\, C \text{ is a cell at level } k\}$, where $\mathrm{diam}(C, S_{\mathcal{X}})$ is the *data-diameter* of the sample points in cell C, defined as $\mathrm{diam}(C, S_{\mathcal{X}}) = \max_{x,y \in C \cap S_{\mathcal{X}}} \|x - y\|$. The diameter of a cell is always an upper bound on its data-diameter.

Theorem 64. *Let $\mathcal{X} = [0,1]^d$ be the domain, $P_{\mathcal{X}}$ a distribution over $\mathcal{X}$, $l : \mathcal{X} \to \{0,1\}$ a labeling function that is ϕ-Lipschitz for some function ϕ, let $q_k = \frac{k \cdot 2 \cdot \ln(2) + \ln(1/\delta)}{\epsilon}$ denote the query numbers of PLAL for level k and let $(\lambda_k)_{k \in \mathbb{N}}$ be a decreasing sequence with $\lambda_k \in [0, \sqrt{d}]$. Then the expected number of queries that PLAL makes on an unlabeled i.i.d. sample $S_{\mathcal{X}}$ from $P_{\mathcal{X}}$ of size m, given that the data diameter of $S_{\mathcal{X}}$ at level k satisfies $\lambda_k^{S_{\mathcal{X}}} \leq \lambda_k$ for all k, is bounded by*

$$\min_{k \in \mathbb{N}} \; (q_k 2^k + \phi(\lambda_k) \cdot m).$$

Proof. For each level, the Probabilistic Lipschitzness allows us to bound the number of points that lie in heterogeneous cells at level k: For any sample point x that lies in a label heterogeneous cluster at level k, there is a sample point y in this cluster, such that the labeling function l on x and y violates the (standard) Lipschitz condition for $1/\lambda_k^{S_\mathcal{X}}$, and thus also for $1/\lambda_k$. The total weight of such points x is bounded by $\phi(\lambda_k)$. Therefore (as λ_k was fixed before drawing the sample), the expected number of sample points that lie in heterogeneous clusters at level k is bounded by $\phi(\lambda_k) \cdot m$. Thus, the expected number of points that are still unlabeled at the beginning of round $k+1$ is bounded by $\phi(\lambda_k) \cdot m$.

Consider the partition of the space PLAL has produced at the beginning of round k (some of the cells in this partition are homogeneous cells from previous rounds and some are the active cells at this level k). Clearly, q_k is a bound on the number of label-queries the algorithm made so far for each of the cells in this partition, as we reuse labels from previous rounds, and the sequence $(q_i)_{i\in\mathbb{N}}$ is non-decreasing. There are at most 2^k cells in this partition. Thus $q_k 2^k$ is an upper bound on the number of queries made up to level k. These two bounds together imply that the number of queries is bounded by $q_k 2^k + \phi(\lambda_k) \cdot m$ for any k. □

The following corollary will allow us to obtain concrete bounds on the number of queries for various probabilistic Lipschitz functions (see Table 6.1 below). It follows directly from Theorem 64. Note that, provided the sequence $(q_i)_{i\in\mathbb{N}}$ of query numbers is non-decreasing, the condition $\phi(\lambda_{k^*}) \cdot m \leq q_{k^*} \cdot 2^{k^* d}$ in the corollary is satisfied for sufficiently large k^*: $\phi(\lambda)$ is decreasing for $\lambda \to 0$, and $\lambda_k \to 0$ for $k \to \infty$ (see comment after Definition 18).

Corollary 65. *Under the conditions of Theorem 64, let k^* be such that $\phi(\lambda_{k^*})\cdot m \leq q_{k^*} \cdot 2^{k^*}$. Then the expected number of queries that PLAL makes on an unlabeled* i.i.d. *sample from $P_\mathcal{X}$ of size m is bounded by*

$$2 \cdot q_{k^*} \cdot 2^{k^*} \;=\; \frac{k^* \cdot 2 \cdot \ln(2) + \ln(1/\delta)}{\epsilon} \cdot 2^{k^*+1}.$$

6.4.4 Bounds for dyadic trees

Here we provide concrete bounds on the expected number of queries for *dyadic trees*. In a dyadic spatial tree, cells are always partitioned by halving one of the coordinates, cycling through the dimensions. That is, for any k, the initial unit cube $[0,1]^d$ (at the root of the tree) is split into 2^{kd} cubes of sidelength $1/2^k$ at level $k \cdot d$. The diameter of such a cube at level kd is $\lambda_{kd} = \sqrt{d}/2^k$, which is at the same time an upper bound on the data diameter $\lambda_{kd}^{S_\mathcal{X}}$ at level kd for any sample $S_\mathcal{X}$.

Table 6.1 provides an overview on the bounds that we get from Corollary 65 for the polynomial and the exponential Lipschitz assumption. For each of the considered probabilistic Lipschitz functions, we first calculate a value k^* such that $\phi(\lambda_{k^*}) \cdot m \leq q_{k^*} \cdot 2^{k^*}$ and then plug this into the formula of Corollary 65 in order to bound the expected number of queries. The calculations can be found below. The simplifications to Landau-notation omit log-factors. We implicitly assume here that the size m of the input sample depends polynomially on $1/\epsilon$, that is $m = O(\frac{1}{\epsilon^\alpha})$ for some $\alpha \in \mathbb{R}$.

Table 6.1: Dyadic trees

Lipschitzness	Bound on expected number of queries
$\phi(\lambda) = \lambda^n$	$2 \cdot \frac{\log(\sqrt{d}^n m\epsilon)^{\frac{d}{n+d}} \ln(2)+\ln(1/\delta)}{\epsilon} \cdot (\sqrt{d}^n m\epsilon)^{\frac{d}{n+d}} = O(m^{\frac{d}{n+d}} \left(\frac{1}{\epsilon}\right)^{\frac{n}{n+d}})$
$\phi(\lambda) = \mathrm{e}^{\frac{-1}{\lambda}}$	$\frac{\sqrt{d}^d \log(\epsilon m)^d}{\epsilon} 2(\log(\log((\epsilon m)^{\sqrt{d}}))d\ln(2) + \ln(2/\delta)) = O(\frac{1}{\epsilon})$

Proofs for the bounds in Table 6.1:

Polynomial Lipschitzness Assume $\phi(\lambda) = \lambda^n$. We determine a k^* such that

$$\phi(\lambda_{k^*}) \cdot m \leq q_{k^*} 2^{k^*},$$

where $q_k = \frac{k \cdot 2\ln(2)+\ln(1/\delta)}{\epsilon}$. Note that, if this inequality holds for some some value $k = k^*$ it will also hold for all $k \geq k^*$. We have $\lambda_{kd} = \frac{\sqrt{d}}{2^k}$. We show that for

$$k = \log(\sqrt{d}^n m\epsilon)^{\frac{1}{n+d}}$$

we have

$$\phi(\lambda_{kd}) \cdot m \leq q_{kd} 2^{kd}.$$

With the above value for k we get

$$k = \frac{\log(\sqrt{d}^n m\epsilon)}{n+d},$$

thus

$$2^{k(n+d)} = \sqrt{d}^n m\epsilon,$$

thus

$$2^{kd}\frac{1}{\epsilon} \geq \frac{\sqrt{d}^n}{2^{kn}}m,$$

thus

$$2^{kd}\frac{kd \cdot 2\ln(2) + \ln(1/\delta)}{\epsilon} \geq \frac{\sqrt{d}^n}{2^{kn}}m,$$

which is what we needed to show. Thus we can set $k^* = kd = d\log(\sqrt{d}^n m\epsilon)^{\frac{1}{n+d}} = \log(\sqrt{d}^n m\epsilon)^{\frac{d}{n+d}}$. According to Corollary 65 the number of queries is now bounded by

$$\begin{aligned} & 2 \cdot \frac{k^* \cdot 2\ln(2) + \ln(1/\delta)}{\epsilon} \cdot 2^{k^*} \\ = \; & 2 \cdot \frac{\log(\sqrt{d}^n m\epsilon)^{\frac{d}{n+d}} \ln(2) + \ln(1/\delta)}{\epsilon} \cdot (\sqrt{d}^n m\epsilon)^{\frac{d}{n+d}} \end{aligned}$$

Exponential Lipschitzness Assume $\phi(\lambda) = \mathrm{e}^{-\frac{1}{\lambda}}$. Again, we determine a k^* such that

$$\phi(\lambda_{k^*}) \cdot m \leq q_{k^*} 2^{k^*},$$

where $q_k = \frac{k \cdot 2\ln(2) + \ln(1/\delta)}{\epsilon}$. We show that for

$$k = \log(\log(\epsilon m)^{\sqrt{d}})$$

we have

$$\phi(\lambda_{kd}) \cdot m \leq q_{kd} 2^{kd}.$$

With the above value for k we get

$$\frac{2^k}{\sqrt{d}} = \log(\epsilon m),$$

thus

$$kd + \frac{2^k}{\sqrt{d}} \log(e) \geq \log(\epsilon m),$$

thus

$$2^{kd} \mathrm{e}^{\frac{2^k}{\sqrt{d}}} \geq \epsilon m,$$

thus

$$(kd \cdot 2\ln(2) + \ln(1/\delta)) \cdot 2^{kd} \mathrm{e}^{\frac{2^k}{\sqrt{d}}} \geq \epsilon m,$$

thus

$$\frac{(kd \cdot 2\ln(2) + \ln(1/\delta))}{\epsilon} \cdot 2^{kd} \geq \mathrm{e}^{\frac{-2^k}{\sqrt{d}}} m,$$

which is what we needed to show. Thus we can set $k^* = kd = d\log(\log(\epsilon m)^{\sqrt{d}})$. According to Corollary 65 the number of queries is now bounded by

$$\begin{aligned} & 2 \cdot \frac{k^* \cdot 2\ln(2) + \ln(1/\delta)}{\epsilon} \cdot 2^{k^*} \\ = \; & \frac{2(\log(\log(\epsilon m)^{\sqrt{d}})d\ln(2) + \ln(1/\delta))}{\epsilon} \cdot (\log(\epsilon m)^{\sqrt{d}})^d \\ = \; & \frac{\sqrt{d}^d \log(\epsilon m)^d}{\epsilon} 2(\log(\log((\epsilon m)^{\sqrt{d}}))d\ln(2) + \ln(1/\delta)). \end{aligned}$$

Remark 66 (Other spatial trees). Often, the *intrinsic dimension* of real data is considerably smaller than the Euclidean dimension of its feature space. Verma et al. [2009] show (for several notions of intrinsic dimension) that, for various classes of spatial trees, the expected data diameter decreases as a function of this intrinsic dimension. Thus, we expect that the query bounds of PLAL used with these trees scale well with the intrinsic dimension.

6.5 PLAL as a Pre-Procedure to Passive Learners

In this section, we argue that using PLAL with dyadic trees as a pre-procedure can reduce the label complexity of a passive learner. In Section 6.5.1, we first show that Empirical Risk Minimizers (ERM algorithms) and Regularized Loss Minimizers (RLM algorithms) are robust to the label errors that PLAL might introduce. This implies that for these types of algorithms it is *safe* to use PLAL for labeling, in the sense that it will not increase the error of the learned classifier by much (and using PLAL can never increase the number of label queries made). Generalizing this, we then argue that it is safe to use labels from PLAL to mimic the oracle for any statistical learning algorithm. In a second step in Section 6.5.2, we prove that there are scenarios, where employing PLAL reduces the label complexity of a learning task.

6.5.1 Robustness of algorithms

In the previous section we have shown how, given any sample, $S = ((x_1, y_1), \dots, (x_m, y_m))$, the PLAL labeling procedure takes its unlabeled projection $S_{\mathcal{X}}(x_1, \dots, x_m)$ as input, queries some of the labels and outputs a labeled sample $S' = ((x_1, y_1'), \dots, (x_m, y_m'))$ such that, with high probability, the number of label errors $|\{i : y_i \neq y_i'\}|$ is bounded (as a function of the Probabilistic Lipschitzness and the number of labels PLAL queried). We show that in many cases such a sample S' suffices for successful learning.

Note that we can not simply invoke Lemma 26 since running PLAL does not give us access to a full labeling function $f : \mathcal{X} \to \{0, 1\}$ of error smaller than ϵ, such that the PLAL-labels could be considered labels from this low-error function. In cells, where PLAL queries all points, we have no information about the labeling of the cell outside the sample points. We thus consider the following notion of robustness of learners:

Definition 67. Let $\mathcal{X}$ be a domain set and P a distribution over $\mathcal{X} \times \{0, 1\}$. Given a sequence of labeled instances, $S = ((x_1, y_1), \dots, (x_m, y_m)) \subseteq \mathcal{X} \times \{0, 1\}$ and $\epsilon \geq 0$, define the ϵ-neighborhood of S as

$$\mathcal{N}_\epsilon(S) = \{S' = ((x_1, y_1'), \dots, (x_m, y_m')) : \ |\{i : y_i \neq y_i'\}|/m \leq \epsilon\}.$$

We say that a learning algorithm $\mathcal{A}$ is $(m, \epsilon, \delta, \eta)$- robust with respect to P, if

$$\Pr_{S \sim P^m} [\ \forall S' \in \mathcal{N}_\epsilon(S),\ \mathrm{Err}_P(\mathcal{A}(S')) \leq \mathrm{Err}_P(\mathcal{A}(S)) + \eta] \geq (1 - \delta).$$

The next lemma (that upper bounds the error introduced by the use of PLAL for robust algorithms) follows directly from this definition and Theorem 62.

Lemma 68. *Let $\mathcal{A}$ be a learner that is $(m, \epsilon, \delta, \eta)$-robust with respect to a distribution P over $[0,1]^d \times \{0,1\}$. Then on random training samples of size m generated by P, replacing the fully labeled sample with one actively labeled by the PLAL (with parameters ϵ, δ), results in deterioration of the error of $\mathcal{A}(S)$ by at most η (with probability at least $(1-2\delta)$ over the samples).*

Next, we show that many common learning algorithms are indeed robust with respect to any data generating distribution, for sufficiently large sample sizes. Applying Lemma 68, we then conclude that for such algorithms PLAL can be applied as a preliminary procedure, and reduce the label complexity of learning, in cases where the query numbers required by PLAL are sufficiently small (so that it compensates for the η loss of accuracy). We require the following basic notions:

Definition 69. Let $\mathcal{X}$ be a domain set, P a distribution over $\mathcal{X} \times \{0,1\}$ and $H \subseteq \{0,1\}^{\mathcal{X}}$ a hypothesis class. We say that a labeled sample S is *ϵ-representative* of H with respect to P, if

$$|\mathrm{Err}_S(h) - \mathrm{Err}_P(h)| \leq \epsilon$$

for every $h \in H$. We say that a class H satisfies the *uniform convergence property* with rate $m_H^{UC} : (0,1) \times (0,1) \to \mathbb{N}$ if, for any data generating distribution P, any $\epsilon, \delta > 0$, and any $m \geq m_H^{UC}(\epsilon, \delta)$, we have

$$\Pr_{S \sim P^m}[S \text{ is } \epsilon\text{-representative for } H \text{ with respect to } P] \geq 1 - \delta.$$

It is well-known that every class H of finite VC-dimension satisfies the uniform convergence property and that there exists a constant c such that, for every such H we have $m_H^{UC}(\epsilon, \delta) = c\frac{\mathrm{VCdim}(H) + \log(1/\delta)}{\epsilon^2}$.

Recall that an algorithm $\mathcal{A}$ is an *Empirical Risk Minimizer (ERM)* for a class H if $\mathcal{A}(S) \in \mathrm{argmin}_{h \in H} \mathrm{Err}_S(h)$. A *Regularized Loss Minimizer (RLM)* $\mathcal{B}$ minimizes a combination of the empirical error and some regularization function $\varphi : H \to \mathbb{R}$, that is $\mathcal{B}(S) \in \mathrm{argmin}_{h \in H}(\mathrm{Err}_S(h) + \varphi(h))$. The following lemma is a consequence of the above definitions.

Lemma 70. *Let $\mathcal{X}$ be a domain set, P a distribution over $\mathcal{X} \times \{0,1\}$ and $H \subseteq \{0,1\}^{\mathcal{X}}$ a hypothesis class. If $m \geq m_H^{UC}(\epsilon, \delta)$ and $\mathcal{A}$ is an ERM (or RLM) algorithm for H, then $\mathcal{A}$, is $(m, \epsilon, \delta, 4\epsilon)$-robust ($(m, \epsilon, \delta, 6\epsilon)$-robust respectively) with respect P.*

Proof. **ERM algorithms:** For some sample S, we let h_S denote an empirical risk minimizer in H with respect to S, that is, $h_S = \text{argmin}_{h \in H} \text{Err}_S(h)$. By Definition 67 we need to show that

$$\Pr_{S \sim P^m} [\ \forall S' \in \mathcal{N}_\epsilon(S),\ \text{Err}_P(h_{S'})) \leq \text{Err}_P(h_S) + 4\epsilon] \geq (1 - \delta).$$

By the Definition 69 (uniform convergence property) we know that a sample of size at least $m \geq m_H^{UC}(\epsilon, \delta)$ is ϵ-representative for H with probability at least $1 - \delta$. Thus, we now assume that the sample S is ϵ-representative and it remains to show that we have for all $S' \in \mathcal{N}(S)$:

$$\text{Err}_P(h_{S'}) \leq \text{Err}_P(h_S) + 4\epsilon.$$

We have

$$\begin{array}{rcll} \text{Err}_P(h_{S'}) & \leq & \text{Err}_S(h_{S'}) + \epsilon & \text{as } S \text{ is } \epsilon\text{-representative} \\ & \leq & \text{Err}_{S'}(h_{S'}) + 2\epsilon & \text{as } S' \in \mathcal{N}(S) \\ & \leq & \text{Err}_{S'}(h_S) + 2\epsilon & \text{by definition of } h_{S'} \\ & \leq & \text{Err}_S(h_S) + 3\epsilon & \text{as } S' \in \mathcal{N}(S) \\ & \leq & \text{Err}_P(h_S) + 4\epsilon & \text{as } S \text{ is } \epsilon\text{-representative} \end{array}$$

RLM algorithms: Now, for some sample S, we let h_S denote a regularized risk minimizer in H with respect to S, that is, $h_S = \text{argmin}_{h \in H}(\text{Err}_S(h) + \varphi(h))$. Again, we assume that the sample S is ϵ-representative and now need to show that we have for all $S' \in \mathcal{N}(S)$:

$$\text{Err}_P(h_{S'}) \leq \text{Err}_P(h_S) + 6\epsilon$$

We start by proving that

$$\varphi(h_S) - \varphi(h_{S'}) \leq 2\epsilon \qquad (*)$$

By way of contradiction, let us assume that, on the contrary, $\varphi(h_S) > \varphi(h_{S'}) + 2\epsilon$. Then we get

$$\begin{array}{rcll}
\mathrm{Err}_S(h_S) + \varphi(h_S) & > & \mathrm{Err}_S(h_S) + \varphi(h_{S'}) + 2\epsilon & \text{by assumption} \\
& \geq & \mathrm{Err}_{S'}(h_S) + \varphi(h_{S'}) + \epsilon & \text{as } S' \in \mathcal{N}(S) \\
& \geq & \mathrm{Err}_{S'}(h_{S'}) + \varphi(h_{S'}) + \epsilon & \text{by definition of } h_{S'} \\
& \geq & \mathrm{Err}_S(h_{S'}) + \varphi(h_{S'}) & \text{as } S' \in \mathcal{N}(S)
\end{array}$$

This contradicts the definition of h_S. With this, we conclude:

$$\begin{array}{rcll}
\mathrm{Err}_P(h_{S'}) & \leq & \mathrm{Err}_S(h_{S'}) + \epsilon & \text{as } S \text{ is } \epsilon\text{-representative} \\
& \leq & \mathrm{Err}_{S'}(h_{S'}) + 2\epsilon & \text{as } S' \in \mathcal{N}(S) \\
& \leq & \mathrm{Err}_{S'}(h_S) + (\varphi(h_S) - \varphi(h_{S'})) + 2\epsilon & \text{by definition of } h_{S'} \\
& \leq & \mathrm{Err}_{S'}(h_S) + 4\epsilon & \text{by } (*) \\
& \leq & \mathrm{Err}_S(h_S) + 5\epsilon & \text{as } S' \in \mathcal{N}(S) \\
& \leq & \mathrm{Err}_P(h_S) + 6\epsilon & \text{as } S \text{ is } \epsilon\text{-representative}
\end{array}$$

□

6.5.2 Reducing labeled sample complexity with PLAL

Table 6.1 provides an upper bound on the expected number of label queries the PLAL procedure makes using dyadic trees, given the unlabeled projection $S_{\mathcal{X}}$ of a sample S of size m, to generate a sample $S' \in \mathcal{N}_\epsilon(S)$. We now apply these bounds to prove reductions in the label complexity achieved by using PLAL as a pre-procedure to passive learning algorithms. Given a passive learning algorithm $\mathcal{A}$, we let $\mathcal{A} \circ \mathrm{PLAL}$ denote the composition of $\mathcal{A}$ with the PLAL procedure. That is, $\mathcal{A} \circ \mathrm{PLAL}$ considers an unlabeled sample $S_{\mathcal{X}}$, applies PLAL to $S_{\mathcal{X}}$ and then applies $\mathcal{A}$ to the resulting labeled sample $S' \in \mathcal{N}_\epsilon(S)$.

Since the PLAL query bounds are derived under the assumption that the data-generating distribution satisfies PL, a fair comparison requires contrasting these with the lower bounds for the sample complexity in the standard model (of learning from fully labeled random training samples) under the same PL assumptions.

In this section, we consider PL-functions ϕ with $\phi(1) = 1$, in particular the "polynomial PL functions", $\phi(\lambda) = \lambda^n$. In this case, the expected number of queries is bounded by $O(m^{\frac{d}{n+d}} \left(\frac{1}{\epsilon}\right)^{\frac{n}{n+d}})$, see Table 6.1. For an algorithm with (fully supervised) sample complexity $m = \Theta((1/\epsilon)^\alpha)$, this yields a $O(\left(\frac{1}{\epsilon}\right)^{\frac{n+\alpha d}{n+d}}))$ bound on the expected number of queries. Thus, using PLAL reduces the label complexity whenever $\alpha > 1$.

PLAL for proper class learning

We start by considering *proper learning*, that is learning a hypothesis class H under the additional requirement that the output classifier is a member of H. Any algorithm that is an ERM or an RLM learner is also a proper learner and we have seen in the previous section that we can use labels from PLAL for these.

Theorem 71. *Let $\mathcal{X} = [0,1]^d$, let $d, n, v \in \mathbb{N}$ and let $\phi(\lambda) = \lambda^n$. Then, there is a hypothesis class H of VC-dimension v, such that for any passive proper learner $\mathcal{A}$, $m^{\text{act}}[\text{PLAL} \circ \text{ERM}, H, \mathcal{Q}^d_{\phi,\text{det}}] = O(\left(\frac{1}{\epsilon}\right)^{\frac{n+2d}{n+d}})$, but $m[\mathcal{A}, H, \mathcal{Q}^d_{\phi,\text{det}}] = \Omega(\frac{1}{\epsilon^2})$, and thus*

$$m^{\text{act}}[\text{PLAL} \circ \text{ERM}, H, \mathcal{Q}^d_{\phi,\text{det}}] = o(m[\mathcal{A}, H, \mathcal{Q}^d_{\phi,\text{det}}]).$$

Proof. We apply the query bound from Table 64 to the sample complexity of $O(1/\epsilon^2)$ of passive proper learning to obtain the upper bound $m^{\text{act}}[\text{PLAL} \circ \text{ERM}, H, \mathcal{Q}^d_{\phi,\text{det}}] = O(\left(\frac{1}{\epsilon}\right)^{\frac{n+2d}{n+d}})$. Comparing this with the lower of $\Omega(\frac{1}{\epsilon^2})$ bound from Theorem 23 yields the result. □

Note that the upper bound on the labeled sample complexity that we obtain for proper learning with PLAL is below the upper bound for proper learning with our SSL framework (see Section 4.5.3 in Chapter 4). The upper bound of $O\left(\left(\frac{1}{\epsilon}\right)^{\frac{d+n}{n}}\right)$ derived there, is obtained by first using a labeled sample to learn a low error classifier, then using this to label an unlabeled sample and learning a classifier from H with this. This could be viewed as a "naive activising strategy", where the preprocedure chooses a subset of the examples uniformly at random and labels the rest of the sample points by their Nearest Neighbors on this subsample. PLAL improving over this "naive strategy" is consistent with our intuition that the ability to choose where to query allows the learner to save labels.

PLAL for unrestricted class learning

Next, we analyze the use of PLAL for unrestricted learning a hypothesis class of finite VC-dimension.

Theorem 72. *For every $n, v \geq 2$ and $d \geq 3n+1$, there exists a class H over $[0,1]^d$ such that $\mathrm{VCdim}(H) = v$ and, for every passive learner $\mathcal{A}$,*

$$m^{\mathrm{act}}[\mathrm{PLAL} \circ \mathrm{ERM}, H, \mathcal{Q}^d_{\phi,\mathrm{det}}] = o(m[\mathcal{A}, H, \mathcal{Q}^d_{\phi,\mathrm{det}}]).$$

We use the following lemma in the proof:

Lemma 73. *Let $0 < \epsilon, \delta < 1/4$, let $\mathcal{X}$ be a domain of size at least $1/\epsilon^3$ and let $\mathcal{Q}$ be the set of distributions over $\mathcal{X} \times \{0,1\}$ whose marginal distribution $P_{\mathcal{X}}$ is uniform over $\mathcal{X}$ and whose labeling function deterministically labels a $(1/2 - \epsilon)$-fraction of the points 0 and $(1/2+\epsilon)$-fraction of the points 1, or the other way around. Let H be the hypothesis class that contains only the constant function 1 and the constant function 0. Then, $(\epsilon/2, \delta)$-learning H with respect to $\mathcal{Q}$ requires a sample size of $\Omega(1/\epsilon^2)$.*

Proof. For every distribution P in $\mathcal{Q}$ we have $\mathrm{opt}_P(H) = 1/2 - \epsilon$. Consider the majority algorithm $\mathcal{M}$ that, given a sample $S = ((x_1, y_1) \ldots (x_m, y_m))$, predicts with a function that agrees with the labels of the sample points on S and outside the sample predicts with the majority label in S. We will now first argue that, for every distribution $P \in \mathcal{Q}$, this algorithm needs to see $\Omega(1/\epsilon^2)$ many points to succeed at the task. Then we show that for any other learning algorithm $\mathcal{A}$, there exits a distribution in $\mathcal{Q}$ where $\mathcal{A}$ performs worse than $\mathcal{M}$. These two steps together imply the claim.

Step 1: Assume that the sample size is $|S| \leq \frac{1}{2\epsilon^2}$. Note that this corresponds to at most an $\epsilon/2$-fraction of the sample points. Thus, if $\mathcal{M}$ predicts (outside of S) with a label that is not the overall (true) majority label, then the error of $\mathcal{M}(S)$ is at least

$1/2 + \epsilon - |S|/|\mathcal{X}| \geq 1/2 + \epsilon/2 > \mathrm{opt}_P(H) + \epsilon/2$. This implies that, for $\mathcal{M}$, $(\epsilon/2, \delta)$-learning H with respect to $\mathcal{Q}$ reduces to correctly learning what the majority label is, that is, it reduces to correctly predicting the bias of a coin. The lower bound in Lemma 5.1 by Anthony and Bartlett [1999] now implies that $\mathcal{M}$ requires a sample larger than $\frac{1}{2\epsilon^2}$ for $\epsilon, \delta < 1/4$.

Step 2: Consider some algorithm $\mathcal{A}$ and assume that this algorithm $(\epsilon/2, \delta)$-learns H with respect to $\mathcal{Q}$ with samples of size m. Fix a sequence of m domain points $(x_1, \ldots, x_m)$. We now consider the expected performance of the learner $\mathcal{A}$ averaged over all distributions in $\mathcal{Q}$, given that the domain points in the sample are $S_{\mathcal{X}} = (x_1, \ldots, x_m)$. Recall that every distribution in $\mathcal{Q}$ has uniform marginal over $\mathcal{X}$, thus the different distributions are distinguished solely by their labeling functions. Slightly abusing the notation, we denote this set of labeling functions also by $\mathcal{Q}$.

Consider a test point x that is not one of the $(x_1, \ldots, x_m)$. Note that, for a fixed labeling of the points in $S_{\mathcal{X}}$, among the labeling functions of distributions in $\mathcal{Q}$ agreeing with that labeling on $S_{\mathcal{X}}$, there are more functions that label x with the majority label on $S_{\mathcal{X}}$ than functions that label x with the minority label on $S_{\mathcal{X}}$. For a labeling function $l \in \mathcal{Q}$, we let S_l denote the points in $S_{\mathcal{X}}$ labeled with l. This implies that

$$\mathbb{E}_{x\sim P_{\mathcal{X}}}\ \mathbb{E}_{l\sim\mathcal{Q}}[\mathbb{1}_{\mathcal{A}(S_l)(x)\neq l(x)} \ \mid\ x \notin S_{\mathcal{X}}] \ \geq\ \mathbb{E}_{x\sim P_{\mathcal{X}}}\ \mathbb{E}_{l\sim\mathcal{Q}}[\mathbb{1}_{\mathcal{M}(S_l)(x)\neq l(x)} \ \mid\ x \notin S_{\mathcal{X}}],$$

where l is chosen uniformly at random from the set $\mathcal{Q}$. As the expectation is commutative, we get

$$\mathbb{E}_{l\sim\mathcal{Q}}\mathbb{E}_{x\sim P_{\mathcal{X}}}[\mathbb{1}_{\mathcal{A}(S_l)(x)\neq l(x)} \ \mid\ x \notin S_{\mathcal{X}}] \ \geq\ \mathbb{E}_{l\sim\mathcal{Q}}\mathbb{E}_{x\sim P_{\mathcal{X}}}[\mathbb{1}_{\mathcal{M}(S_l)(x)\neq l(x)} \ \mid\ x \notin S_{\mathcal{X}}].$$

As this is independent of the choice of $S_{\mathcal{X}}$, we further obtain

$$\mathbb{E}_{S_{\mathcal{X}}\sim\mathcal{P}^m}\mathbb{E}_{l\sim\mathcal{Q}}\mathbb{E}_{x\sim P_{\mathcal{X}}}[\mathbb{1}_{\mathcal{A}(S_l)(x)\neq l(x)} \ \mid\ x \notin S_{\mathcal{X}}] \ \geq\ \mathbb{E}_{S_{\mathcal{X}}\sim\mathcal{P}^m}\mathbb{E}_{l\sim\mathcal{Q}}\mathbb{E}_{x\sim P_{\mathcal{X}}}[\mathbb{1}_{\mathcal{M}(S_l)(x)\neq l(x)} \ \mid\ x \notin S_{\mathcal{X}}].$$

This yields

$$\mathbb{E}_{l\sim\mathcal{Q}}\mathbb{E}_{S_{\mathcal{X}}\sim\mathcal{P}^m}\mathbb{E}_{x\sim P_{\mathcal{X}}}[\mathbb{1}_{\mathcal{A}(S_l)(x)\neq l(x)} \ \mid\ x \notin S_{\mathcal{X}}] \ \geq\ \mathbb{E}_{l\sim\mathcal{Q}}\mathbb{E}_{S_{\mathcal{X}}\sim\mathcal{P}^m}\mathbb{E}_{x\sim P_{\mathcal{X}}}[\mathbb{1}_{\mathcal{M}(S_l)(x)\neq l(x)} \ \mid\ x \notin S_{\mathcal{X}}].$$

This implies that there exists a function $l \in \mathcal{Q}$ such that

$$\mathbb{E}_{S_{\mathcal{X}}\sim\mathcal{P}^m}\mathbb{E}_{x\sim P_{\mathcal{X}}}[\mathbb{1}_{\mathcal{A}(S_l)(x)\neq l(x)} \ \mid\ x \notin S_{\mathcal{X}}] \ \geq\ \mathbb{E}_{S_{\mathcal{X}}\sim\mathcal{P}^m}\mathbb{E}_{x\sim P_{\mathcal{X}}}[\mathbb{1}_{\mathcal{M}(S_l)(x)\neq l(x)} \ \mid\ x \notin S_{\mathcal{X}}].$$

That is, for this distribution with labeling function l, the expected error of $\mathcal{A}$ is larger than the expected error of $\mathcal{M}$ (outside the sample). This completes the proof of the lemma. $\square$

With this we can proceed to the proof of the above theorem.

Proof of Theorem 72. We consider the class $H = \{f_{i,j} : i, j \in \{0,1\}\}$ of functions that are constant on $\mathcal{X} \setminus \bar{0}$. More precisely, we define $f_{i,j}$ to be the function with $f(\bar{0}) = i$ and $f(x) = j$ for all $x \neq \bar{0}$. This is a class of VC-dimension 2. We show that for every $\epsilon < 1/4$ there exist a class of distributions $\mathcal{Q}_\epsilon \subseteq \mathcal{Q}^d_{\phi,\mathrm{det}}$ such that passively learning the class H with respect to $\mathcal{Q}_\epsilon$ requires a sample size of $\Omega(\frac{1}{\epsilon^{1.5}})$, whereas applying PLAL allows us to learn H with only $O(\frac{1}{\epsilon})$ many queries.

For $\mathcal{Q}_\epsilon$, we consider all distributions that have support $\{\bar{0}\} \cup G$, where G is a grid of sidelength $\phi^{-1}(\sqrt{\epsilon})$ such that every point in G has distance at least 1 from $\bar{0}$. As in the proof of Theorem 22, we can construct such a grid with $d\left(\frac{1}{\phi^{-1}(\sqrt{\epsilon})}\right)^{d-1} = d\left(\frac{1}{\epsilon}\right)^{\frac{d-1}{2n}} \geq \left(\frac{1}{\epsilon}\right)^{1.5}$ many points (where the last inequality follows from $d \geq 3n+1$). All distributions in $\mathcal{Q}_\epsilon$ assign weight $1 - \sqrt{\epsilon}$ to $\bar{0}$ and distribute the remaining weight $\sqrt{\epsilon}$ uniformly over G. We further allow all labeling functions that assign a $(1/2 - \sqrt{\epsilon})$-fraction of the gridpoints one label (either 0 or 1) and a $(1/2 + \sqrt{\epsilon})$-fraction of the gridpoints the other label (and assign any label to $\bar{0}$). By construction, each of these distributions is ϕ-Lipschitz.

The approximation error of the class H is $\mathrm{Err}_P(H) = \sqrt{\epsilon}(\frac{1}{2} - \sqrt{\epsilon}) = (\frac{\sqrt{\epsilon}}{2} - \epsilon)$ for every distribution $P \in \mathcal{Q}_\epsilon$. Thus, ϵ-learning the class H with respect to $\mathcal{Q}_\epsilon$ corresponds to $\sqrt{\epsilon}$-learning the class that contains only the constant 1 and the constant 0 functions on the grid points. Since $|G| \geq \frac{1}{\epsilon^{1.5}}$, Lemma 73 implies that $\Omega\left(\frac{1}{\epsilon}\right)$ many sample points on the grid are necessary for this task. By Theorem 16, $O\left(\frac{1}{\epsilon}\right)$ are also sufficient. As the total weight of the gridpoints is $\sqrt{\epsilon}$, a random sample from the distribution needs to be of size $\Omega(\frac{1}{\epsilon^{1.5}})$, for that many hits to the grid. However, it is easy to see that PLAL would only query the label of $\bar{0}$ once. Thus, using PLAL for this task, results in labeled sample complexity $O(\frac{1}{\epsilon})$. $\square$

PLAL for Nearest Neighbor learning

Finally, we analyze the use of PLAL for Nearest Neighbor learning. Lemma 68 does not imply that Nearest Neighbor is a robust algorithm. In order to show, that using PLAL can also reduce the label complexity of unrestricted learning, we consider a slight variant of the standard 1-NN algorithm and denote this by $\mathrm{NN} \circ \mathrm{PLAL}$. Instead of labeling each point by the label of its nearest neighbor in the space, we consider the partition of the space into cells at the end of the run of PLAL, and label each point with the label of its nearest neighbor *within its cell*. If a point falls into a cell that is empty, we label it with the label of its nearest neighbor *within its parent-cell* (note that this one is never empty). This slight modification allows us to show the following:

We adapt the proof from Theorem 20 for the success of the 1-Nearest Neighbor algorithm under Lipschitzness to its modified version of 1-NN with PLAL. We will here prove the following:

Lemma 74. *Let P be a distribution over $[0,1]^d$ with PL-function $\phi(\lambda) = \lambda^n$. Then applying* $\mathrm{NN} \circ \mathrm{PLAL}$ *to an unlabeled sample $S_{\mathcal{X}}$ of size*

$$m \geq \left(\frac{1}{\epsilon}\right)^{\frac{d}{n}+1} \frac{(2\sqrt{d})^d}{\delta \mathrm{e}}$$

results in classification error at most 2ϵ with probability at least $(1-\delta)$ (over the choice of $S_{\mathcal{X}}$).

Proof. Let $\lambda = \sqrt{d}/2^k$ for the smallest k such that $\sqrt{d}/2^k \leq \phi^{-1}(\epsilon)$. This implies $\phi^{-1}(\epsilon) \geq \lambda \geq \phi^{-1}(\epsilon)/2$. We can cover $\mathcal{X} = [0,1]^d$ with $r = \left(\sqrt{d}/\lambda\right)^d = 2^{kd}$ boxes $C_1, C_2, \ldots C_r$ of side-length $\lambda/\sqrt{d} = 1/2^k$. Note that any two points inside such a box are at distance at most λ.

Using Markov's inequality, Lemma 19 from Chapter 3 implies that for any $\epsilon > 0$ and m we have

$$\Pr_{S \sim P^m}\left[\left[\sum_{i:C_i \cap S = \emptyset} P[C_i]\right] > \epsilon\right] \leq \frac{r}{\epsilon m e}$$

It follows that in this setting, for any $\epsilon, \delta > 0$, a sample of size

$$m \geq \left(\frac{\sqrt{d}}{\lambda}\right)^d \frac{1}{\epsilon\delta \mathrm{e}} = \frac{r}{\epsilon\delta\mathrm{e}}$$

suffices to guarantee that with probability exceeding $(1-\delta)$, at most an ϵ-fraction of domain points are in boxes that are not hit by the sample. By noting that $\phi^{-1}(\epsilon) = \epsilon^{1/n}$ (for the polynomial Lipschitzness function $\phi(\lambda) = \lambda^n$) and recalling that $\lambda \geq \phi^{-1}(\epsilon)/2$, we obtain that

$$\left(\frac{2\sqrt{d}}{\epsilon^{1/n}}\right)^d \frac{1}{\epsilon\delta\mathrm{e}} = \left(\frac{2\sqrt{d}}{\phi^{-1}(\epsilon)}\right)^d \frac{1}{\epsilon\delta\mathrm{e}} \geq \left(\frac{\sqrt{d}}{\lambda}\right)^d \frac{1}{\epsilon\delta\mathrm{e}} = \frac{r}{\epsilon\delta\mathrm{e}}.$$

Therefore, the sample size stated above suffices for hitting all but an ϵ-fraction of the boxes.

Now consider the modified 1-NN labeling rule, where every point x gets the label of its Nearest Neighbor within the cell that a run of PLAL produced on a sample $S_{\mathcal{X}}$. We denote the sample $S_{\mathcal{X}}$ with the labels from PLAL by S . We refer to the elements of the partition that PLAL produced as *cells* and to the elements of the partition in the argument above as *boxes*. All these elements are axis-alligned rectangles that have powers of 1/2 as sidelengths. For a point x, we denote the box that contains x by $b(x)$ and the cell that contains x by $c(x)$. As the sidelengths of both boxes and cells are powers of 1/2, and we use the dyadic spatial trees, we have $b(x) \subset c(x)$ or $c(x) \subset b(x)$ or $b(x) = c(x)$ for all x.

To bound the probability that a test point x receives the wrong label, we consider the following cases:

Case 1: $c(x)$ was declared homogeneous by PLAL.
Then x will receive the label of $c(x)$. By Remark 63, the total error resulting restricted to this condition is at most ϵ.

Case 2: $c(x)$ was not declared homogeneous by PLAL and $b(x) \subseteq c(x)$.
We chose the sample size of S so that (with probability at least $1-\delta$) at most an ϵ-fraction of points lie in boxes that are not hit by S, thus the probability (over the choice of x) that $S \cap b(x) = \emptyset$ is bounded by ϵ. If $S \cap b(x) \neq \emptyset$, then the Nearest Neighbor of x inside $c(x)$ has distance at most λ from x (recall that the diameter of $b(x)$ is λ). As $\phi^{-1}(\epsilon) \geq \lambda$, at most an ϵ fraction of points x are at distance less than λ from some point of opposite label. Thus, the error of our labeling rule in this case is at most 2ϵ.

Case 3: $c(x)$ was not declared homogeneous by PLAL, $c(x) \subset b(x)$ and $c(x) \cap S \neq \emptyset$. We can bound the probability that x receives a wrong label by 2ϵ in the same way as in Case 2. (The probability that $b(x) \cap S = \emptyset$ is bounded by ϵ and otherwise x receives the label of a point that is at distance at most λ.)

Case 4: $c(x)$ was not declared homogeneous by PLAL, $c(x) \subset b(x)$ and $c(x) \cap S = \emptyset$. In this case x receives the label of its Nearest Neighbor in the parent cell of $c(x)$. We denote this cell by $p(c(x))$. The cell $c(x)$ was produced when PLAL decided to split $p(c(x))$. Thus the parent cell $p(c(x))$ contains points from S. Note that $c(x) \subset b(x)$ implies $p(c(x)) \subseteq b(x)$. This implies that the Nearest Neighbor of x in $p(c(x))$ is at distance at most λ from x and as under Case 2 we bound the probability that this neighbor has a different label than x by ϵ. □

Now the lower bound from Theorem 22 in Chapter 3 together with the above result implies:

Theorem 75. *Let $d, n \geq 2$ and let $\phi(\lambda) = \lambda^n$. Applying* PLAL *to the Nearest Neighbor algorithm (in the way described above) results in active sample complexity for learning $\mathcal{Q}_\phi$ that is below the sample complexity of any passive learning algorithm for that class. Namely, for any passive learner $\mathcal{A}$, $m[\mathcal{A}, \mathcal{Q}_\phi] = \Omega((\frac{1}{\epsilon})^{1+\frac{d-1}{n}})$, but $m^{\mathrm{act}}[\mathrm{NN} \circ \mathrm{PLAL}, \mathcal{Q}_\phi] = O((\frac{1}{\epsilon})^{1+\frac{d^2}{n(n+d)}})$, and thus*

$$m^{\mathrm{act}}[\mathrm{NN} \circ \mathrm{PLAL}, \mathcal{Q}_\phi] = o(m[\mathcal{A}, \mathcal{Q}_\phi]).$$

Proof. For $\phi(\lambda) = \lambda^n$, the lower bound for unrestricted learning in Theorem 22 becomes $\Omega((\frac{1}{\epsilon})^{1+\frac{d-1}{n}})$. If we apply $\mathrm{NN} \circ \mathrm{PLAL}$ with samples of size $\Theta((\frac{1}{\epsilon})^{\frac{d+n}{n}})$ (see Lemma 74), we reduced the label complexity to $O((\frac{1}{\epsilon})^{1+\frac{d^2}{n(n+d)}})$ (note that $\frac{d^2}{n(n+d)} \leq \frac{d-1}{n}$ for any $d, n \geq 2$). □

Summary of reductions

Our upper bounds for active learning with PLAL together with the lower bounds for learning under Probabilistic Lipschitzness in Chapter 3 imply the following reductions in labeled sample complexity:

Table 6.2: Label complexity reductions

	Passive	PLAL-Active
Proper Learning of H	$\Omega\left(1/\epsilon^2\right)$	$O\left(\left(\frac{1}{\epsilon}\right)^{\frac{n+2d}{n+d}}\right)$
Unrestricted Learning of H	$\Omega\left(\frac{1}{\epsilon^{1.5}}\right)$	$O\left(\frac{1}{\epsilon}\right)$
Nearest Neighbor Learning	$\Omega\left(\left(\frac{1}{\epsilon}\right)^{1+\frac{d-1}{n}}\right)$	$O\left(\left(\frac{1}{\epsilon}\right)^{1+\frac{d^2}{n(n+d)}}\right)$

6.6 Discussion

In this section, we have presented a procedure, PLAL, that takes an unlabeled sample (and accuracy and confidence parameters) as input, and outputs a full labeling of the sample points while querying the label of only a fraction of these points. We have shown that we can bound the number of label queries that PLAL makes in terms of the Probabilistic Lipschitzness of the data-generating distribution. This lead to reductions in labeled sample complexity for proper learning, unrestricted class learning and Nearest Neighbor learning under PL in comparison to these learning scenarios in the (passive) standard framework.

By analyzing PLAL under the assumption of Probabilistic Lipschitzness, we have introduced a new measure to the literature of active learning theory. Intuitively, the possibility to actively choose which points to label should be beneficial to a learner, if there are dense label homogenous areas, where usual *i.i.d.* sampling would “waste” labels due to this high density, while an active learner would need to make only one (or few) queries to identify the label of this dense area. Our analysis of active learning under Probabilistic Lipschitzness provides a way to formalize and solidify this intuition. Most previous work has provided bounds in terms of the disagreement coefficient, which is a measure that relates the hypothesis class to be learned to the target distribution. Probabilistic Lipschitzness, in contrast, is a property of the underlying data distribution only and we thereby obtain bounds that are independent of a specific hypothesis class.

Using PLAL as a preprocedure to a passive learning algorithm, can be viewed as “avtivising” this passive learner. In contrast to previously suggested activising procedures, PLAL is computationally efficient. The number of cells it produces is at most the number of points in the input sample, the running time is thus polynomial in the size of the input sample.

In this chapter, we have shown that using PLAL is safe for ERM and RLM learners, since labels from PLAL do not change the empirical error of any classifier by much. Urner et al. [2013] argue, more generally, that it is safe to use PLAL to to mimic the input to any *statistical algorithms*. The framework of learning from statistical queries has been introduced independently by Ben-David et al. [1990] (as "learning by distances") and by Kearns [1993]. Feldman et al. [2013] shows that many practically relevant learners can be cast in this framework.

Finally, Urner et al. [2013] also present some experimental results on using PLAL for some synthetic datasets. They illustrate that using PLAL label queries results in higher accuracy (of NN) than choosing a random subset of the same size for label query. In addition, they show that PLAL is extensible in a straightforward manner to multi-label learning settings.

Chapter 7

Concluding Remarks

We have analyzed machine learning paradigms that are frequently and successfully employed in practice, but are not appropriately modeled by standard learning theory. In particular, this thesis presents formal results on the usefulness of unlabeled data and data that is generated by a different task than the target task. We hope that our analysis advances the understanding of when learning with such training data is possible and beneficial.

The data assumption of Probabilistic Lipschitzness has proven to be meaningful for the analysis of all three non-standard learning settings investigated in this thesis. As we have argued earlier, data assumptions are necessary for any provable results on the benefits of unlabeled data. This is implied by lower bounds that were previously established in the standard worst-case (over all distributions) framework of analysis. In practice however, these lower bounds often prove to be too pessimistic. In order to understand this discrepancy, we propose to analyze machine learning paradigms under data assumptions that aim to model natural learning tasks (as opposed to learning tasks that are constructed to defeat a worst case framework of analysis). In particular, we suggest that learning theory put a greater focus on exploring how to model (parameters of) naturalness of learning tasks. We believe that our analysis of non-standard learning settings under Probabilistic Lipschitzness provides a step in this direction.

References

Micah Adler and Brent Heeringa. Approximating optimal binary decision trees. *Algorithmica*, 62(3-4):1112–1121, 2012.

Martin Anthony and Peter L. Bartlett. *Neural Network Learning: Theoretical Foundations.* Cambridge University Press, 1999.

Maria-Florina Balcan and Avrim Blum. A pac-style model for learning from labeled and unlabeled data. In *Proceedings of the Conference on Learning Theory (COLT)*, pages 111–126, 2005.

Maria-Florina Balcan, Andrei Z. Broder, and Tong Zhang. Margin based active learning. In *Proceedings of the Conference on Learning Theory (COLT)*, pages 35–50, 2007.

Tugkan Batu, Lance Fortnow, Ronitt Rubinfeld, Warren D. Smith, and Patrick White. Testing closeness of discrete distributions. *J. ACM*, 60(1):4, 2013.

Shai Ben-David and Ami Litman. Combinatorial variability of vapnik-chervonenkis classes with applications to sample compression schemes. *Discrete Applied Mathematics*, 86(1): 3–25, 1998.

Shai Ben-David and Ruth Urner. On the hardness of domain adaptation and the utility of unlabeled target samples. In *Proceedings of the Conference on Algorithmic Learning Theory (ALT)*, pages 139–153, 2012.

Shai Ben-David and Ruth Urner. Domain adaptation–can quantity compensate for quality? To appear in the Annals of Mathematics and Artificial Intelligence (AMAI), 2013.

Shai Ben-David, Alon Itai, and Eyal Kushilevitz. Learning by distances. In *Proceedings of the Conference on Learning Theory (COLT)*, pages 232–245, 1990.

Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. Analysis of representations for domain adaptation. In *Advances in Neural Information Processing Systems (NIPS)*, pages 137–144, 2006.

Shai Ben-David, Tyler Lu, and Dávid Pál. Does unlabeled data provably help? worst-case analysis of the sample complexity of semi-supervised learning. In *Proceedings of the Conference on Learning Theory (COLT)*, pages 33–44, 2008.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine Learning*, 79(1-2):151–175, 2010a.

Shai Ben-David, Tyler Lu, Teresa Luu, and Dávid Pál. Impossibility theorems for domain adaptation. *Journal of Machine Learning Research - Proceedings Track*, 9:129–136, 2010b.

Shai Ben-David, Shai Shalev-Shwartz, and Ruth Urner. Domain adaptation–can quantity compensate for quality? In *International Symposium on Artificial Intelligence and Mathematics (ISAIM)*, 2012.

Shalev Ben-David and Shai Ben-David. Learning a classifier when the labeling is known. In *Proceedings of the Conference on Algorithmic Learning Theory (ALT)*, pages 440–451, 2011.

Alina Beygelzimer, Sanjoy Dasgupta, and John Langford. Importance weighted active learning. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 49–56, 2009.

Alina Beygelzimer, Daniel Hsu, John Langford, and Tong Zhang. Agnostic active learning without constraints. In *Advances in Neural Information Processing Systems (NIPS)*, pages 199–207, 2010.

Anselm Blumer, Andrzej Ehrenfeucht, David Haussler, and Manfred K. Warmuth. Learnability and the vapnik-chervonenkis dimension. *J. ACM*, 36(4):929–965, 1989.

Cristian Bucila, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *Proceedings of the International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 535–541, 2006.

Vittorio Castelli and Thomas M. Cover. On the exponential value of labeled samples. *Pattern Recognition Letters*, 16(1):105–111, 1995.

Vittorio Castelli and Thomas M. Cover. The relative value of labeled and unlabeled samples in pattern recognition with an unknown mixing parameter. *IEEE Transactions on Information Theory*, 42(6):2102–2117, 1996.

O. Chapelle, B. Schölkopf, and A. Zien, editors. *Semi-Supervised Learning*. MIT Press, Cambridge, MA, 2006.

Olivier Chapelle and Alexander Zien. Semi-supervised classification by low density separation. In *Proceedings of the Tenth International Workshop on Articial Intelligence and Statistics*, pages 57–64, 2005.

Rita Chattopadhyay, Shayok Chakraborty, Vineeth Nallure Balasubramanian, and Sethuraman Panchanathan. Optimization-based domain adaptation towards person-adaptive classification models. In *ICMLA (1)*, pages 476–483, 2011.

Minmin Chen, Kilian Q. Weinberger, and John Blitzer. Co-training for domain adaptation. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2456–2464, 2011.

Corinna Cortes and Mehryar Mohri. Domain adaptation in regression. In *Proceedings of the Conference on Algorithmic Learning Theory (ALT)*, pages 308–323, 2011.

Corinna Cortes, Mehryar Mohri, Michael Riley, and Afshin Rostamizadeh. Sample selection bias correction theory. In *Proceedings of the Conference on Algorithmic Learning Theory (ALT)*, pages 38–53, 2008.

Corinna Cortes, Yishay Mansour, and Mehryar Mohri. Learning bounds for importance weighting. In J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R.S. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems (NIPS)*, pages 442–450. 2010.

Sanjoy Dasgupta. Analysis of a greedy active learning strategy. In *Advances in Neural Information Processing Systems (NIPS)*, pages 337–344. MIT Press, 2004.

Sanjoy Dasgupta. Coarse sample complexity bounds for active learning. In *Advances in Neural Information Processing Systems (NIPS)*, 2005.

Sanjoy Dasgupta. Two faces of active learning. *Theor. Comput. Sci.*, 412(19):1767–1781, 2011.

Sanjoy Dasgupta and Daniel Hsu. Hierarchical sampling for active learning. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 208–215, 2008.

Sanjoy Dasgupta, Wee Sun Lee, and Philip M. Long. A theoretical analysis of query selection for collaborative filtering. *Machine Learning*, 51(3):283–298, 2003.

Sanjoy Dasgupta, Daniel Hsu, and Claire Monteleoni. A general agnostic active learning algorithm. In *International Symposium on Artificial Intelligence and Mathematics (ISAIM)*, 2008.

Hal Daumé III and Jagadeesh Jagarlamudi. Domain adaptation for machine translation by mining unseen words. In *Association for Computational Linguistics*, Portland, OR, 2011.

Vitaly Feldman, Elena Grigorescu, Lev Reyzin, Santosh Vempala, and Ying Xiao. Statistical algorithms and a lower bound for detecting planted cliques. In *STOC*, pages 655–664, 2013.

Alon Gonen, Sivan Sabato, and Shai Shalev-Shwartz. Efficent active learning of halfspaces: an aggressive approach. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2013.

Boqing Gong, Yuan Shi, Fei Sha, and Kristen Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2066–2073, 2012.

Steve Hanneke. A bound on the label complexity of agnostic active learning. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 353–360, 2007.

Steve Hanneke. Activized learning: Transforming passive to active with improved label complexity. *Journal of Machine Learning Research (JMLR)*, 13(May):1469–1587, 2012.

D Haussler and E Welzl. Epsilon-nets and simplex range queries. In *Proceedings of the second annual symposium on Computational geometry*, SCG '86, pages 61–71, 1986.

Jiayuan Huang, Arthur Gretton, Bernhard Schölkopf, Alexander J. Smola, and Karsten M. Borgwardt. Correcting sample selection bias by unlabeled data. In *Advances in Neural Information Processing Systems (NIPS)*. MIT Press, 2007.

Matti Kääriäinen. Generalization error bounds using unlabeled data. In *Proceedings of the Conference on Learning Theory (COLT)*, pages 127–142, 2005.

Matti Kääriäinen. Active learning in the non-realizable case. In *Proceedings of the Conference on Algorithmic Learning Theory (ALT)*, pages 63–77, 2006.

Michael J. Kearns. Efficient noise-tolerant learning from statistical queries. In *Proceedings of the ACM Symposium on the Theory of Computing (STOC)*, pages 392–401, 1993.

Benjamin G. Kelly, Thitidej Tularak, Aaron B. Wagner, and Pramod Viswanath. Universal hypothesis testing in the learning-limited regime. In *IEEE International Symposium on Information Theory (ISIT)*, pages 1478–1482, 2010.

Daniel Kifer, Shai Ben-David, and Johannes Gehrke. Detecting change in data streams. In *VLDB*, pages 180–191, 2004.

Percy Liang, Hal Daumé III, and Dan Klein. Structure compilation: trading structure for features. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 592–599, 2008.

Enno Mammen and Alexandre B. Tsybakov. Smooth discrimination analysis. *Annals of Statistics*, 27(6):1808–1829, 1999.

Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain adaptation: Learning bounds and algorithms. In *Proceedings of the Conference on Learning Theory (COLT)*, 2009.

Pascal Massart and Élodie Nédélec. Risk bounds for statistical learning. *Annals of Statistics*, 34(5):2326–2366, 2006.

Swarnajyoti Patra and Lorenzo Bruzzone. A cluster-assumption based batch mode active learning technique. *Pattern Recognition Letters*, 33(9):1042–1048, 2012.

Philippe Rigollet. Generalized error bounds in semi-supervised classification under the cluster assumption. *Journal of Machine Learning Research (JMLR)*, 8:1369–1392 (electronic), 2007.

Burr Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning*. 2014. To be published by Cambridge University Press.

Yuan Shi and Fei Sha. Information-theoretical learning of discriminative clusters for unsupervised domain adaptation. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2012.

Aarti Singh, Robert D. Nowak, and Xiaojin Zhu. Unlabeled data: Now it helps, now it doesn't. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1513–1520, 2008.

Ingo Steinwart and Andreas Christmann. *Support Vector Machines*. Springer Publishing Company, Incorporated, 1st edition, 2008.

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

M. Sugiyama and K. Mueller. Generalization error estimation under covariate shift. In *Workshop on Information-Based Induction Sciences*, 2005.

M. Sugiyama, M. Krauledat, and K.-R. Muller. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8:985–1005, 2007.

Masashi Sugiyama, Taiji Suzuki, Shinichi Nakajima, Hisashi Kashima, Paul von Bünau, and Motoaki Kawanabe. Direct importance estimation for covariate shift adaptation. *Annals of the Institute of Statistical Mathematics*, 60(4):699–746, 2008.

Y. Tsuboi, Kashima, Hido S. H., S. Bickel, and M Sugiyama. Direct density ratio estimation for large-scale covariate shift adaptation. *Journal of Information Processing*, 17:138–155, 2009.

Alexandre B. Tsybakov. Optimal aggregation of classifiers in statistical learning. *Annals of Statistics*, 32(1):135–166, 2004.

Ruth Urner, Shai Ben-David, and Shai Shalev-Shwartz. Unlabeled data can speed up prediction time. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 641–648, 2011.

Ruth Urner, Shai Ben-David, and Ohad Shamir. Learning from weak teachers. *Journal of Machine Learning Research - Proceedings Track*, 22:1252–1260, 2012.

Ruth Urner, Sharon Wullf, and Shai Ben-David. Plal: Cluster-based active learning. In *Proceedings of the Conference on Learning Theory (COLT)*, 2013.

Leslie G. Valiant. A theory of the learnable. *Commun. ACM*, 27(11):1134–1142, 1984.

Vladimir N. Vapnik and Alexey J. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability & Its Applications*, 16 (2):264–280, 1971.

Nakul Verma, Samory Kpotufe, and Sanjoy Dasgupta. Which spatial partition trees are adaptive to intrinsic dimension? In *UAI*, pages 565–574, 2009.

Liwei Wang. Smoothness, disagreement coefficient, and the label complexity of agnostic active learning. *Journal of Machine Learning Research*, 12:2269–2292, 2011.

Xiaojin Zhu. Semi-supervised learning literature survey, 2008. URL `http://pages.cs.wisc.edu/~jerryzhu/research/ssl/semireview.html`.