Reliable Agnostic Learning

Adam Tauman Kalai *
Microsoft Research
Cambridge, MA
adum@microsoft.com

Varun Kanade †
Georgia Tech
Atlanta, GA
varunk@cc.gatech.edu

Yishay Mansour ‡
Google Research and
Tel Aviv University
mansour@tau.ac.il

Abstract

It is well-known that in many applications erroneous predictions of one type or another must be avoided. In some applications, like spam detection, false positive errors are serious problems. In other applications, like medical diagnosis, abstaining from making a prediction may be more desirable than making an incorrect prediction. In this paper we consider different types of *reliable classifiers* suited for such situations. We formalize and study properties of reliable classifiers in the spirit of agnostic learning (Haussler, 1992; Kearns, Schapire, and Sellie, 1994), a PAC-like model where no assumption is made on the function being learned. We then give two algorithms for reliable agnostic learning under natural distributions. The first reliably learns DNF formulas with no false positives using membership queries. The second reliably learns halfspaces from random examples with no false positives or false negatives, but the classifier sometimes abstains from making predictions.

1 Introduction

In many machine learning applications, a crucial requirement is that mistakes of one type or another should be avoided at all cost. As a motivating example, consider spam detection, where classifying a correct email as spam (a false positive) can have dire, or even fatal consequences. On the other hand, if a spam message is not tagged correctly (a false negative) it is comparatively a smaller problem. In other situations, abstaining from making a prediction might be better than making wrong predictions; in a medical test it is preferable to have inconclusive predictions to wrong ones, so that the remaining predictions can be relied upon. A reliable classifier would have a pre-specified error bound for false positives or false negatives or both. We present formal models for different types of *reliable classifiers* and give efficient algorithms for some of the learning problems that arise.

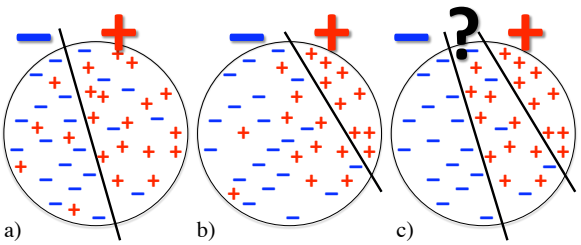

Figure 1: Three different learning settings. Depending on the application and data, one may be more appropriate than the others. (a) The best (most accurate) classifier (from the class of halfspaces) for a typical agnostic learning problem. (b) The best positive-reliable halfspace classifier for a problem like spam-prediction in which false positives are to be avoided. (c) The best fully reliable halfspace sandwich classifier for a problem in which all errors are to be avoided. It predicts −, +, or ?.

In agnostic learning [Hau92, KSS94], one would like provably efficient learning algorithms that make no assumption about the function to be learned. The learner's goal is to nearly match (within ϵ) the accuracy of the best classifier from a specified class of functions (See Figure 1a). Agnostic learning can be viewed as PAC learning [Val84] with arbitrary noise. In reliable agnostic learning, the learner's goal is to output a nearly reliable classifier whose accuracy nearly matches the accuracy of the best *reliable* classifier from a specified class of functions (See Figure 1b-c).

Since agnostic learning is extremely computationally demanding, it is interesting that we can find efficient algorithms for reliably agnostic learning interesting classes of functions over natural distributions. Our

*Part of this research was done while the author was at Georgia Institute of Technology, supported in part by NSF SES-0734780, and NSF CAREER award, and a SLOAN Fellowship

†The author was supported in part by NSF CCF-0746550

‡This work was supported in part by the IST Programme of the European Community, under the PASCAL2 Network of Excellence, by a grant from the Israel Science Foundation and by a grant from United States-Israel Binational Science Foundation (BSF). This publication reflects the authors' views only.

algorithms build on recent results in agnostic learning, one for learning decision trees [GKK08] and one for learning halfspaces [KKMS05]. Throughout the paper, our focus is on *computationally efficient* algorithms for learning problems. The contributions of this paper are the following:

- We introduce a model of reliable agnostic learning. Following prior work on agnostic learning, we consider both distribution-specific and distribution-free learning, as well as both learning from membership queries and from random examples. We show that reliable agnostic learning is no harder that agnostic learning and no easier that PAC learning.

- We give an algorithm for reliably learning DNF (with almost no false positives) over the uniform distribution, using membership queries. More generally, we show that if concept class $\mathcal{C}$ is agnostically learnable, then the class of disjunctions of concepts from $\mathcal{C}$ is reliably learnable (with almost no false positives).

- We give an algorithm for reliably learning halfspace sandwich classifiers over the unit ball under uniform distribution. This algorithm is fully reliable in the sense that it almost never makes mistakes of any type. We also extend this algorithm to have *tolerant reliability*, in which case a permissible rate of false positives and false negatives is specified, and the goal of the algorithm is to achieve maximal accuracy subject to these constraints.

Positive Reliability: A *positive-reliable* classifier is one that never produces false positives. (One can similarly define a *negative-reliable* classifier as one that never produces false negatives.) On the spam example, a positive-reliable DNF could be

$$(\texttt{Nigeria} \wedge \texttt{bank} \wedge \texttt{transaction} \wedge \texttt{million}) \vee$$
$$(\texttt{Viagra} \wedge \neg \texttt{COLT} \ldots .$$

An email should be almost certainly spam if it fits into any of a number of categories, where each category is specified by a sets of words that it must contain and must not contain.

Our goal is to output a classifier that is (almost) positive-reliable and has false negative rate (almost) as low as the best positive-reliable classifier from a fixed set of classifiers, such as size-s DNFs. Consider a positive-reliable classifier that has least rate of false negatives. We require our algorithm to output a classifier that has rate of false negatives (within ϵ) as low as this best one, and require that the false positive rate of our classifier be less than ϵ.

Our first algorithm efficiently learns the class of DNF formulas in the positive reliable agnostic model over the uniform distribution and uses membership queries. Our model is conceptually akin to a one-sided error agnostic noise model; hence this may be a step towards agnostically learning DNF. Note that our algorithm also gives an alternative way of learning DNFs (without noise) that is somewhat different than Jackson's celebrated harmonic sieve [Jac97].

More generally, we show that if a class of functions is efficiently agnostically learnable, then polynomial-sized disjunctions of concepts from that class are efficiently learnable in the positive reliable agnostic model. Similarly, it can be shown that agnostically learning a class of functions implies learning polynomial-size conjunctions of concepts from that class in the negative reliable agnostic model. A consequence of this is that a polynomial-time algorithm for agnostically learning DNFs over the uniform distribution would give an algorithm for the challenging problem uniform-PAC learning depth-3 circuits (since PAC learning is easier than reliable learning).

Full Reliability: We consider the notion of *full-reliability*, which means simultaneous positive and negative reliability. In order to achieve this, we need to consider *partial classifiers*, ones that may predict positive, negative, or "?", where a "?" means no prediction. A partial classifier is fully reliable if it never produces false positives or false negatives. Given a concept class of partial classifiers, the goal is to find a (nearly) fully reliable classifier that is almost as accurate as the best fully reliable classifier from the concept class.

We show that reliable agnostic learning is easier (or no harder) than agnostic learning; if a concept class is efficiently learnable in the agnostic setting, it is also efficiently learnable in the positive-reliable, negative-reliable, and full-reliable models.

Tolerant reliability: We also consider the notion of tolerant reliability, where we are willing to tolerate given rates τ_+, τ_- of false positives and false negatives. In this most general version, we consider the class of halfspace sandwich partial classifiers (halfspace sandwiches for short), in which the examples that are in one halfspace are positive, examples that are in another are negative, and the rest of the examples are classified as "?". We extend the agnostic halfspace algorithm and analysis of Kalai, Klivans, Mansour, and Servedio [KKMS05] to the case of halfspace sandwiches. In particular, we show that given arbitrary rates τ_+, τ_-, we can learn the class of halfspace sandwiches to within any constant ϵ over the uniform distribution in polynomial time using random examples. Our algorithm outputs a hypothesis h such that h has false positive and false negative rates close (within ϵ) to τ_+ and τ_- respectively and has accuracy within ϵ of the best halfspace sandwich with false positive and false negative rates bounded by τ_+, τ_-.

Related Work: A classical approach to the problem of reliable classification is defining a loss function that has different penalties for different type of errors. For example, by having an infinite loss on false positive errors and a loss of one on false negative errors, we essentially define a positive-reliable classifier as one that minimizes the loss. The main issue that arises is computational, since there is no efficient way to compute a halfspace that would minimize a general loss function. The contribution of this paper is to show that one can define interesting sub-classes for which the computational tasks

are feasible.

Conceptually, our work is similar to the study of cautious classifiers, i.e. classifiers which may output "unknown" as a label[FHO04]. The models we introduce, in particular the full-reliable learning model is very similar to the bounded-improvement model in [Pie07]. The methods we use to prove the reduction between agnostic learning and reliable learning are related to work on delegating classifiers [FFHO04] and cost-sensitive learning [Elk01, ZLA03].

The *cost-sensitive classification* model is a more general model than reliable learning. In particular, to simulate positive reliability, one would impose a very large cost on false positives. However, we do not see how to extend our DNF algorithm to this setting. Even when the mistake costs are equal, the problem of agnostically learning DFN remains an open problem [GKK08]. The requirement that there be almost no false positives seems to make the problem significantly simpler. Extending our halfspace algorithm to the cost-sensitive classification setting seems more straightforward.

The motivation of our work is highly related to the Neyman-Pearson criterion, an application of which is the following: given a joint distribution over points and labels, minimize the rate of false negatives subject to the condition that the rate of false positives is bounded by a given input parameter. The Neyman-Pearson criterion classifies points based on the ratio between the likelihood of the point being labeled positive and negative. Neyman and Pearson [NP33] proved that the optimal classification strategy is to choose a threshold on the ratio of the likelihoods. This method was applied to statistical learning in [CHHS02, SN05], to solve constrained version of empirical risk minimization or structural risk minimization. Unfortunately, the optimization problems that arise for most classes of interest are computationally intractable. In contrast, our work focuses on deriving computationally efficient algorithms for some interesting concept classes.

The main focus of our work is to develop formal models for reliable learning and give algorithms with theoretical guarantees on their performance. Our models are similar in spirit to Valiant's PAC learning model [Val84] and agnostic learning models by Kearns, Schapire and Sellie [KSS94]. In our algorithms we do not change the underlying distribution on the instance space, but sometimes flip labels of instances. This allows us to convert distribution-specific agnostic learning algorithms into algorithms for reliable learning.

Organization. For readability, the paper is divided into three parts. In section 2, we focus only on positive reliability. Section 3 contains a reduction from agnostic learning to fully reliable learning. Finally, in section 4, we consider reliable learning with pre-specified permissible error rates.

2 Positive reliability

The main result we prove in this section is the following: if a concept class $\mathcal{C}$ is efficiently agnostically learnable, the composite class of size-s disjunctions of concepts from $\mathcal{C}$ is efficiently positive-reliably learnable. An application of this result is an algorithm for positive reliably learning size-s DNF formulas, based on the recent result by Gopalan, Kalai and Klivans [GKK08] for agnostically learning decision trees. (Recall that a monomial has a small decision tree and therefore the class of DNF is included in the class of disjunctions of decision trees.)

2.1 Preliminaries

Throughout the paper, we will consider $\langle X_n \rangle_{n \geq 1}$ as the instance space (for example the boolean cube $X_n = \{0,1\}^n$ or the unit ball $X_n = \mathcal{B}_n$). For each n, the target is an arbitrary function $f_n : X_n \to [0,1]$, which we interpret as $f_n(x) = \Pr[y = 1|x]$. A distribution $\mathcal{D}_n$ over X_n together with f_n, induces a joint distribution, $(\mathcal{D}_n, f_n)$, over examples $X_n \times \{0,1\}$ as follows: To draw a random example (x, y), pick $x \in X_n$ according to distribution $\mathcal{D}_n$, set $y = 1$ with probability $f_n(x)$, and otherwise $y = 0$.

A *false positive* is a prediction of 1 when the label is $y = 0$. Similarly a *false negative* is a prediction of 0 when the label is $y = 1$. The rate of false positives and negatives of a classifier $c : X_n \to \{0,1\}$ are defined below. When $\mathcal{D}_n$ and f_n are clear from context, we will omit them.

$$\text{false}_+(c) = \text{false}_+(c, \mathcal{D}_n, f_n) \triangleq \Pr_{(x,y)\sim(\mathcal{D}_n, f_n)}[c(x) = 1 \wedge y = 0] = \mathop{\mathbf{E}}_{x\sim\mathcal{D}_n}[c(x)(1 - f_n(x))]$$

$$\text{false}_-(c) = \text{false}_-(c, \mathcal{D}_n, f_n) \triangleq \Pr_{(x,y)\sim(\mathcal{D}_n, f_n)}[c(x) = 0 \wedge y = 1] = \mathop{\mathbf{E}}_{x\sim\mathcal{D}_n}[(1 - c(x))f_n(x)]$$

A classifier c is said to be *positive-reliable* if $\text{false}_+(c) = 0$. In other words, it never makes false positive predictions. Although, we focus on positive reliability, an entirely similar definition can be made in terms of negative reliability. The *error* of classifier c is,

$$\text{err}(c) = \text{err}(c, \mathcal{D}_n, f_n) \triangleq \Pr_{(x,y)\sim(\mathcal{D}_n, f_n)}[c(x) \neq y] = \text{false}_+(c) + \text{false}_-(c)$$

To keep notation simple, we will drop the subscript n, except in definitions.

Oracles: As is standard in computational learning theory, we consider two types of oracles: membership query (MQ) and example (EX). Given a target function $f : X \to [0,1]$ we define the behavior of the two types of oracles below:

- *Membership Query (MQ) oracle*: For a query $x \in X$, the oracle returns $y = 1$ with probability $f(x)$, and $y = 0$ otherwise, independently each time it is invoked.

- *Example (EX) oracle*: When invoked, the oracle draws $x \in X$ according to the distribution $\mathcal{D}$, sets $y = 1$ with probability $f(x)$, $y = 0$ otherwise and returns (x, y).

Almost all our results hold for both types of oracles and none of our algorithms change the underlying distribution $\mathcal{D}$ over X. We use $\mathcal{O}(f)$ to denote an oracle, which may be of either of these types. Note that neither type of oracle we consider is persistent – either may return $(x,1)$ and $(x,0)$ for the same x. In this sense, our model is similar to the p-concept distribution model for agnostic learning introduced in Kearns, Schapire and Sellie [KSS94]. Whenever an algorithm $\mathcal{A}$ has access to oracle $\mathcal{O}(f)$ we denote it by $\mathcal{A}^{\mathcal{O}(f)}$.

Agnostic Learning. *Algorithm $\mathcal{A}$* efficiently agnostically learns *sequence of concept classes $\langle \mathcal{C}_n \rangle_{n\geq 1}$, under distributions $\langle \mathcal{D}_n \rangle_{n\geq 1}$, if there exists a polynomial $p(n,1/\epsilon,1/\delta)$ such that, for every $n \geq 1, \epsilon, \delta > 0$ and every $f_n : X_n \to [0,1]$, with probability at least $1-\delta$, $\mathcal{A}^{\mathcal{O}(f_n)}(\epsilon,\delta)$ outputs hypothesis h that satisfies*

$$\mathrm{err}(h, \mathcal{D}_n, f_n) \leq \min_{c \in \mathcal{C}_n} \mathrm{err}(c, \mathcal{D}_n, f_n) + \epsilon.$$

The time complexity of both $\mathcal{A}$ and h is bounded by $p(n,1/\epsilon,1/\delta)$.

Below we define positive-reliable learning and postpone definitions of full reliability and tolerant reliability to sections 3 and 4 respectively. Define the subset of positive-reliable classifiers (relative to a fixed distribution and target function), to be: $\mathcal{C}^+ = \{c \in \mathcal{C} \mid \mathrm{false}_+(c) = 0\}$. Note that if we assume that $\mathcal{C}$ contains the classifier which predicts 0 on all examples, then $\mathcal{C}^+$ is non-empty.

Positive Reliable Learning. *Algorithm $\mathcal{A}$* efficiently positive reliably learns *a sequence of concept classes $\langle \mathcal{C}_n \rangle_{n\geq 1}$, under distributions $\langle \mathcal{D}_n \rangle_{n\geq 1}$, if there exists a polynomial $p(n,1/\epsilon,1/\delta)$ such that, for every $n \geq 1$, $\epsilon, \delta > 0$ and every $f_n : X_n \to [0,1]$, with probability at least $1-\delta$, $\mathcal{A}^{\mathcal{O}(f_n)}(\epsilon,\delta)$ outputs hypothesis h that satisfies* $\mathrm{false}_+(h, \mathcal{D}_n, f_n) \leq \epsilon$ *and*

$$\mathrm{false}_-(h, \mathcal{D}_n, f_n) \leq \min_{c \in \mathcal{C}_n^+} \mathrm{false}_-(c, \mathcal{D}_n, f_n) + \epsilon.$$

The time complexity of both $\mathcal{A}$ and h is bounded by $p(n,1/\epsilon,1/\delta)$.

Each oracle call is assumed to take unit time. Hence, an upper bound on the run-time is also an upper-bound on the sample complexity, i.e., the number of oracle calls.

2.2 Positive reliably learning DNF

Our main theorem for this section is the following:

Theorem 1. *Let $\mathcal{A}$ be an algorithm that (using oracle $\mathcal{O}(f)$) efficiently agnostically learns concept class $\mathcal{C}$ under distribution $\mathcal{D}$. There exists an algorithm $\mathcal{A}'$ that (using oracle $\mathcal{O}(f)$ and black-box access to $\mathcal{A}$) efficiently positive reliably learns the class of size-s disjunctions of concepts from $\mathcal{C}$.*

Using the result on learning decision trees [GKK08], we immediately get Corollary 1. The remainder of this section is devoted to proving Theorem 1 and Corollary 1.

Corollary 1. *There is an MQ algorithm $\mathcal{B}$ such that for any $n, s \geq 1$, $\mathcal{B}$ positive reliably learns the class of size-s DNF formulas on n variables in time* $\mathrm{poly}(n, s, \frac{1}{\epsilon}, \frac{1}{\delta})$ *with respect to the uniform distribution.*

We show that if we have access to oracle $\mathcal{O}(f)$, we can simulate oracle $\mathcal{O}(f')$ where $f'(x) = q(x)f(x) + (1-q(x))r(x)$, where $q,\ r : X \to [0,1]$ are arbitrary functions and we only assume black-box access to q and r.

Lemma 1. *Given access to oracle $\mathcal{O}(f)$ and black-box access to the functions q and r we can simulate oracle $\mathcal{O}(f')$ for $f' = qf + (1-q)r$.*

Proof. We show this assuming that $\mathcal{O}(f)$ is an example oracle; the case when $\mathcal{O}(f)$ is a membership query oracle is simpler. Let $f' = qf + (1-q)r$. To simulate $\mathcal{O}(f')$, we do the following: First draw (x,y) from $\mathcal{O}(f)$, with probability $q(x)$, return (x,y). With probability $1-q(x)$ do the following: Set $y' = 1$ with probability $r(x)$, $y' = 0$ otherwise, return (x,y'). It is easy to see that $\Pr[y = 1|x] = f'(x)$, thus this simulates the oracle $\mathcal{O}(f')$. □

When $\mathcal{C}$ is a concept class that is agnostically learnable, we give an algorithm that uses the agnostic learner as a black-box and outputs a hypothesis which has a low rate of false positives. This hypothesis will have rate of false negatives close to optimum with respect to the positive-reliable concepts in $\mathcal{C}$. Our construction modifies the target function f, leaving the distribution $\mathcal{D}$ unchanged, in such a way that false positives (with respect to the original function) are penalized much more than false negatives. By correctly choosing parameters, the output of the black-box agnostic learner is close to the best positive-reliable classifier.

Lemma 2. *Assume that algorithm $\mathcal{A}^{\mathcal{O}(f)}(\epsilon,\delta)$ efficiently agnostically learns concept class $\mathcal{C}$ under distribution $\mathcal{D}$ in time $T(\epsilon,\delta)$. Then, $\mathcal{A}^{\mathcal{O}(f')}(\epsilon^2/2,\delta)$, where $f' = (1/2+\epsilon/4)f$, positive reliably learns $\mathcal{C}$ in time $T(\epsilon^2/2,\delta)$.*

Proof. Let $f' = \left(\frac{1}{2} + \frac{\epsilon}{4}\right) f$. Let $g : X \to \{0,1\}$ be an arbitrary function and let $p_1 = \mathbf{E}_{x\sim\mathcal{D}}[f(x)]$. We relate the quantities false_+ and false_- of g with respect to the functions f and f' - simple calculations show that,

$$\mathrm{false}_+(g,\mathcal{D},f') = \mathrm{false}_+(g,\mathcal{D},f) + \left(\frac{1}{2} - \frac{\epsilon}{4}\right)(p_1 - \mathrm{false}_-(g,\mathcal{D},f)) \quad (1)$$

$$\mathrm{false}_-(g,\mathcal{D},f') = \left(\frac{1}{2} + \frac{\epsilon}{4}\right)\mathrm{false}_-(g,\mathcal{D},f) \quad (2)$$

$$\mathrm{err}(g,\mathcal{D},f') = \mathrm{false}_+(g,\mathcal{D},f) + \left(\frac{1}{2} - \frac{\epsilon}{4}\right)p_1 + \frac{\epsilon}{2}\mathrm{false}_-(g,\mathcal{D},f) \quad (3)$$

Let $c \in \mathcal{C}$ be such that $\mathrm{false}_+(c,\mathcal{D},f) = 0$ and $\mathrm{false}_-(c,\mathcal{D},f) = \mathrm{opt}_+ = \min_{c'\in\mathcal{C}^+}\mathrm{false}_-(c',\mathcal{D},f)$, so that if h is the output of $\mathcal{A}^{\mathcal{O}(f')}(\frac{\epsilon^2}{2},\delta)$, with probability at least $1-\delta$,

$$\mathrm{err}(h,\mathcal{D},f') \leq \mathrm{err}(c,\mathcal{D},f') + \frac{\epsilon^2}{2} \quad (4)$$

Substituting identity (3) in (4), once for c and once for h we get

$$\begin{aligned}\text{false}_+(h,\mathcal{D},f) &\leq \frac{\epsilon}{2}(\text{opt}_+ - \text{false}_-(h,\mathcal{D},f)) + \frac{\epsilon^2}{2} \\ &\leq \epsilon\end{aligned} \tag{5}$$

Dropping the non-negative term $\text{false}_+(h,\mathcal{D},f)$ from (5) and rearranging we get,

$$\begin{aligned}\frac{\epsilon}{2}\,\text{false}_-(h,\mathcal{D},f') &\leq \frac{\epsilon}{2}\text{opt}_+ + \frac{\epsilon^2}{2} \\ \text{false}_-(h,\mathcal{D},f) &\leq \text{opt}_+ + \epsilon\end{aligned} \tag{6}$$

□

For learning disjunction of concepts, we first need to learn the concept class on a subset of the instance space. We show that in the agnostic setting, learning on a subset of the instance space is only as hard as learning over the entire instance space. The simple reduction we present here does not seem feasible in the noiseless case. Let $S \subseteq X$ be a subset, a distribution $\mathcal{D}$ over X induces a conditional distribution $\mathcal{D}|_S$ over S; for any $T \subseteq X$, $\Pr_{\mathcal{D}|_S}[T] = \Pr_{\mathcal{D}}[T \cap S]/\Pr_{\mathcal{D}}[S]$. We assume access to the *indicator function* for the set S, i.e., $\mathrm{I}_S : X \to \{0,1\}$ such that $\mathrm{I}_S(x) = 1$ if $x \in S$ and $\mathrm{I}_S(x) = 0$ otherwise. If $\Pr_{\mathcal{D}}[S]$ is not negligible, it is possible to agnostically learn under conditional distribution $\mathcal{D}|_S$ using a black-box agnostic learner.

Lemma 3. *Suppose algorithm $\mathcal{A}^{\mathcal{O}(f)}(\epsilon,\delta)$ efficiently agnostically learns $\mathcal{C}$ under distribution $\mathcal{D}$ in time $T(\epsilon,\delta)$. Let $S \subseteq X$ such that, $\Pr_{\mathcal{D}}[S] \geq \gamma(= 1/\text{poly})$. Then, algorithm $\mathcal{A}^{\mathcal{O}(f')}(\epsilon\gamma,\delta)$, where $f' = f\mathrm{I}_S + (1-\mathrm{I}_S)/2$, efficiently agnostically learns $\mathcal{C}$ under distribution $\mathcal{D}|_S$ in time $T(\epsilon\gamma,\delta)$.*

Proof. We first relate the errors with respect to f and $f' = f\mathrm{I}_S + (1-\mathrm{I}_S)/2$. In words, f' is f on S and random outside S. Therefore, the error on any function g would be $1/2$ outside of S and its error on $\mathcal{D}|_S$ inside S. More formally, for any function $g : X \to \{0,1\}$ we have,

$$\begin{aligned}\text{err}(g,\mathcal{D},f') &= \mathop{\mathbf{E}}_{x\sim\mathcal{D}}[g(x)(1-f'(x)) + (1-g(x))f'(x)] \\ &= \mathop{\mathbf{E}}_{x\sim\mathcal{D}}[(g(x)(1-f(x)) + (1-g(x))f(x))\mathrm{I}_S] \\ &\quad + \mathop{\mathbf{E}}_{x\sim\mathcal{D}}[(1-\mathrm{I}_S)/2] \\ &= \Pr_{\mathcal{D}}[S]\,\text{err}(g,\mathcal{D}|_S,f) + (1-\Pr_{\mathcal{D}}[S])/2\end{aligned} \tag{7}$$

Let $c \in \mathcal{C}$ be such that $\text{err}(c,\mathcal{D}|_S,f) = \min_{c'\in\mathcal{C}}\text{err}(c',\mathcal{D}|_S,f)$ = opt. Using (7) we can conclude that c is also optimal under distribution $\mathcal{D}$ with respect to function f'. Algorithm $\mathcal{A}^{\mathcal{O}(f')}(\epsilon\gamma,\delta)$ outputs hypothesis h which satisfies with probability at least $1-\delta$, $\text{err}(h,\mathcal{D},f') \leq \text{err}(c,\mathcal{D},f') + \epsilon\gamma$. Using (7) and since $\Pr_{\mathcal{D}}[S] \geq \gamma$ we immediately get $\text{err}(h,\mathcal{D}|_S,f) \leq \text{opt} + \epsilon$. □

We define algorithm CPRL (conditional positive reliable learner) as follows. The input to the algorithm is ϵ - the accuracy parameter, δ - the confidence parameter, γ - bound on the probability of S, $\mathcal{O}(f)$ – oracle to access the target function and I_S the indicator function for subset S. Algorithm CPRL initially defines $f'' = (1/2+\epsilon/4)(f\mathrm{I}_S + (1-\mathrm{I}_S)/2)$. By Lemma 1 we can sample from $\mathcal{O}(f'')$ given access to $\mathcal{O}(f)$ and I_S. Algorithm CPRL runs $\mathcal{A}^{\mathcal{O}(f'')}(\epsilon^2\gamma^2/2,\delta)$ and returns its output. We state the properties of algorithm CPRL as Lemma 4

input: ϵ, δ, M, $\mathcal{O}(f)$, CPRL
set $H_0 := 0$;
for $i = 1$ to M {
  set $m := \frac{2}{\epsilon^2}\log\frac{2M}{\delta}$;
  draw $Z = \langle(x_1,y_1),\ldots,(x_m,y_m)\rangle$ from distribution $(\mathcal{D},f)$;
  set $\hat{p}_i := \sum_{j=1}^{m}(1-H_{i-1}(x_j))$;
  if $\hat{p}_i \geq \frac{\epsilon}{2}$ {
    $h_i :=$ CPRL$(\frac{\epsilon}{2M}, \frac{\delta}{2M}, \frac{2}{3}\hat{p}_i, \mathcal{O}(f), 1-H_{i-1})$;
  } **else** {
    $h_i := 0$;
  }
  $H_i := H_{i-1} \vee h_i$;
}
output H_M

Figure 2: Algorithm Disjunction Learner

Lemma 4. *Suppose algorithm $\mathcal{A}^{\mathcal{O}(f)}(\epsilon,\delta)$ agnostically learns $\mathcal{C}$ under distribution $\mathcal{D}$ in time $T(\epsilon,\delta)$. For a subset $S \subseteq X$ with $\Pr_{\mathcal{D}}[S] \geq \gamma(= 1/\text{poly})$, algorithm* CPRL *(which uses $\mathcal{A}$ as a black-box) with probability at least $1-\delta$ returns a hypothesis h such that* $\text{false}_+(h,\mathcal{D}|_S,f) \leq \epsilon$ *and*

$$\text{false}_-(h,\mathcal{D}|_S,f) \leq \min_{c\in\mathcal{C}^+}\text{false}_-(c,\mathcal{D}|_S,f) + \epsilon$$

Algorithm CPRL *has access to oracle $\mathcal{O}(f)$ and black-box access to I_S. The running time of algorithm* CPRL *is $T(\epsilon^2\gamma^2/2,\delta)$.*

Let $\mathsf{OR}_s(\mathcal{C}) = \{c_1 \vee \cdots \vee c_s \mid c_j \in \mathcal{C}, \forall j \leq s\}$ be the composite class of size-s disjunctions of concepts from $\mathcal{C}$. Theorem 2 states that if $\mathcal{C}$ is agnostically learnable under distribution $\mathcal{D}$, then $\mathsf{OR}_s(\mathcal{C})$ is positive reliably learnable under distribution $\mathcal{D}$.

We need two simple lemmas to prove Theorem 2, whose proofs are elementary and we omit.

Lemma 5. *Let $\mathcal{D}$ be a distribution over X, $f : X \to [0,1]$ an arbitrary function and let c, h be such that* $\text{false}_+(c) = 0$ *and* $\text{false}_-(h) \leq \text{false}_-(c) + \epsilon$, *then* $\Pr_{\mathcal{D}}[h(x)=1] \geq \Pr_{\mathcal{D}}[c(x)=1] - \epsilon$

Lemma 6. *Let $\mathcal{D}$ be a distribution over X, $f : X \to [0,1]$ an arbitrary function and let c, h be such that* $\text{false}_+(c) = 0$, $\text{false}_+(h) \leq \epsilon$ *and* $\Pr_{\mathcal{D}}[h(x)=1] \geq \Pr_{\mathcal{D}}[c(x)=1] - \epsilon$, *then* $\text{false}_-(h) \leq \text{false}_-(c) + \epsilon$

Lemma 5 states that if a hypothesis h has a rate of false negatives close to that of a concept c that predicts

no false positives, then h must predict 1 almost as often as c. Lemma 6 states that if h is a hypothesis with low rate of false positives and if it predicts 1 almost as often as a concept c which never predicts false positives, the rate of false negatives of h must be close to that of c.

Theorem 2. *Let $\mathcal{C}$ be a concept class that is agnostically learnable (in polynomial time) under $\mathcal{D}$ and $f : X \to [0,1]$ be an arbitrary function. Algorithm* DISJUNCTION-LEARNER *(see Fig. 2) run with parameters ϵ, δ, $M = s\log(2/\epsilon)$ and access to oracle $\mathcal{O}(f)$ and black-box access to* CPRL*, with probability at least $1-\delta$ outputs a hypothesis H_M, such that* $\text{false}_+(H_M, \mathcal{D}, f) \leq \epsilon$ *and*

$$\text{false}_-(H_M, \mathcal{D}, f) \leq \min_{\psi \in \mathsf{OR}_s(\mathcal{C})^+} (\psi, \mathcal{D}, f) + \epsilon.$$

The running time is polynomial in $s, \frac{1}{\epsilon}, \frac{1}{\delta}$.

Proof. For definitions of H_i and p_i refer to figure 2. Let $\varphi \in \mathsf{OR}_s(\mathcal{C})$ satisfy $\text{false}_+(\varphi, \mathcal{D}, f) = 0$ and $\text{false}_-(\varphi, \mathcal{D}, f) = \text{opt}_+ = \min_{\psi \in \mathsf{OR}_s(\mathcal{C})^+} \text{false}_-(\psi, \mathcal{D}, f)$. Suppose $\varphi = c_1 \vee \cdots \vee c_s$, where $c_j \in \mathcal{C}$. Let $S_i = \{x \in X \mid H_{i-1}(x) = 0\}$, and hence $1 - H_{i-1}$ is an indicator function for S_i. Let $p_i = \Pr_{\mathcal{D}}[S_i]$ and $p_+ = \Pr_{\mathcal{D}}[\varphi(x) = 1]$. Our goal is to show that $\Pr_{\mathcal{D}}[H_M(x) = 1] \geq p_+ - \epsilon$ and $\text{false}_+(H_M, \mathcal{D}, f) \leq \epsilon$ and then using Lemma 6 we are done.

In the ith iteration, we compute using a sample, $\hat{p}_i$ to estimate p_i. Using Hoeffding's bound, with probability at least $1 - (\delta/2M)$, $|\hat{p}_i - p_i| \leq (\epsilon/2)$. Let us suppose this holds for all M iterations, allowing our algorithm a failure probability of $\delta/2$ so far. In this scenario, if $p_i \geq \epsilon$, $\hat{p}_i \geq (\epsilon/2)$ and $\hat{p}_i \leq (3p_i/2)$. Note that if $p_i < \epsilon$, then $\text{false}_-(H_{i-1}, \mathcal{D}, f) < \epsilon$ and $\Pr_{\mathcal{D}}[H_{i-1}(x) = 1] \geq 1 - \epsilon$ and hence we are done. Let $\mathcal{D}|_{S_i}$ denote the conditional distribution given S_i and in the ith iteration the call to CPRL returns a hypothesis h_i such that $\text{false}_+(h_i, \mathcal{D}|_{S_i}, f) \leq \frac{\epsilon}{2M}$ and $\text{false}_-(h_i, \mathcal{D}|_{S_i}, f) \leq \text{opt}_i + \frac{\epsilon}{2M}$ where $\text{opt}_i = \min_{c \in \mathcal{C}^+} \text{false}_-(c, \mathcal{D}|_{S_i}, f)$. Note that for $j = 1, \ldots, s$, $\text{false}_+(c_j, \mathcal{D}|_{S_i}, f) = 0$, and hence $\text{false}_-(c_j, \mathcal{D}|_{S_i}, f) \geq \text{opt}_i$ for all i. Thus we get,

$$\text{false}_-(h_i, \mathcal{D}|_{S_i}, f) \leq \text{false}_-(c_j, \mathcal{D}|_{S_i}, f) + \frac{\epsilon}{2M}$$

and hence using Lemma 5,

$$\Pr_{\mathcal{D}|_{S_i}}[h_i(x) = 1] \geq \Pr_{\mathcal{D}|_{S_i}}[c_j(x) = 1] - \frac{\epsilon}{2M}$$

$$\Pr_{\mathcal{D}|_{S_i}}[h_i(x) = 1] \geq \frac{1}{s} \Pr_{\mathcal{D}|_{S_i}}[\varphi(x) = 1] - \frac{\epsilon}{2M}$$

Let $q_i = 1 - p_i = \Pr_{\mathcal{D}}[H_i(x) = 1]$. We define the quantity $d_i = p_+ - q_i$ and show by induction that $d_i \leq p_+ \left(1 - \frac{1}{s}\right)^i + \frac{\epsilon}{2M} \sum_{j=0}^{i-1} \left(1 - \frac{1}{s}\right)^j$. To check the base step see that $d_1 = p_+ - q_1 = p_+ - \Pr_{\mathcal{D}}[h_1(x) = 1] \leq p_+ - \frac{p_+}{s} + \frac{\epsilon}{2M}$. For the induction step we have:

$$\begin{aligned}
d_{i+1} &= p_+ - \Pr_{\mathcal{D}}[H_{i+1}(x) = 1] \\
&= p_+ - \Pr_{\mathcal{D}}[H_i(x) = 1] - \Pr_{\mathcal{D}}[H_i(x) = 0 \wedge h_{i+1}(x) = 1] \\
&= d_i - q_{i+1} \Pr_{\mathcal{D}|_{S_{i+1}}}[h_{i+1}(x) = 1] \\
&\leq d_i - \frac{q_{i+1}}{s} \Pr_{\mathcal{D}|_{S_{i+1}}}[\varphi(x) = 1] + \frac{\epsilon}{2M} \\
&\leq d_i - \frac{1}{s}(\Pr_{\mathcal{D}}[\varphi(x) = 1] - \Pr_{\mathcal{D}}[H_i(x) = 1]) + \frac{\epsilon}{2M} \\
&\leq d_i \left(1 - \frac{1}{s}\right) + \frac{\epsilon}{2M} \\
&\leq p_+ \left(1 - \frac{1}{s}\right)^{i+1} + \frac{\epsilon}{2M} \sum_{j=0}^{i} \left(1 - \frac{1}{s}\right)^j
\end{aligned}$$

When $M = s \log \frac{2}{\epsilon}$, $d_M \leq \epsilon$, and it is easily checked that $\text{false}_+(H_M, \mathcal{D}, f) \leq \epsilon$ and hence using Lemma 6, $\text{false}_-(H_M, \mathcal{D}, f) \leq \text{opt}_+ + \epsilon$. The probability of failure some call to CPRL is at most $M \frac{\delta}{2M} = \delta/2$, which combined with probability of failure caused by incorrect estimation of some $\hat{p}_i$ gives total failure probability at most δ. □

Theorem 2 is a more precise restatement of Theorem 1. This combined with Lemma 7 below (which is Theorem 18 from [GKK08]) proves Corollary 1.

Lemma 7. [GKK08] *The class of polynomial-size decision trees can be agnostically learned (using queries) to accuracy ϵ in time* $\text{poly}(n, \frac{1}{\epsilon}, \frac{1}{\delta})$ *with respect to the uniform distribution.*

3 Full reliability

In this section we deal with the case of full reliability. We are interested in obtaining a hypothesis that has low error rate, in terms of false positives and negatives both. In the noisy (agnostic) setting, this is not possible unless we allow our hypothesis to refrain from making a prediction. For this reason we use the notion of *partial classifiers* which predict a value from the set $\{0, 1, ?\}$, where prediction of "?" is treated as uncertainty of the classifier. Recall, that our instance space is $\langle X_n \rangle_{n \geq 1}$ and for each n, the target is an unknown function $f_n : X \to [0,1]$. Formally, a partial classifier is $c : X_n \to \{0, 1, ?\}$. Let $\text{I}(\mathcal{E})$ denote the indicator function for event $\mathcal{E}$. The error (similarly false_+, false_-) of a partial classifier can be defined as

$$\text{err}(c, \mathcal{D}_n, f_n) = \mathop{\mathbf{E}}_{x \sim \mathcal{D}_n}[\text{I}(c(x) = 0) f_n(x) + \text{I}(c(x) = 1)(1 - f_n(x))]$$

We define the *uncertainty* of a partial classifier c to be

$$?(c, \mathcal{D}_n, f_n) = \mathop{\mathbf{E}}_{x \sim \mathcal{D}_n}[\text{I}(c(x) = ?)]$$

Finally we define the *accuracy* of a partial classifier c to be

$$\text{acc}(c, \mathcal{D}_n, f_n) = \mathop{\mathbf{E}}_{x \sim \mathcal{D}_n}[\text{I}(c(x) = 0)(1 - f_n(x)) + \text{I}(c(x) = 1) f_n(x)]$$

Note that for a partial classifier c, we have $\text{err}(c)+?(c)+\text{acc}(c) = 1$.

Suppose that $\langle \mathcal{C}_n \rangle_{n\geq 1}$ is a sequence of concept classes, recall that we defined $\mathcal{C}_n^+ = \{c \in \mathcal{C}_n \mid \text{false}_+(c) = 0\}$. Similarly we define $\mathcal{C}_n^- = \{c \in \mathcal{C}_n \mid \text{false}_-(c) = 0\}$. We can now define the class of partial classifiers derived from $\mathcal{C}_n$ as $\mathsf{PC}(\mathcal{C}_n) = \{(c_+, c_-) \mid c_+ \in \mathcal{C}_n^+,\ c_- \in \mathcal{C}_n^-\}$. The definition ensures that for a partial classifier $c = (c_+, c_-) \in \mathsf{PC}(\mathcal{C}_n)$, with probability 1 a sample $x \sim D_n$ satisfies $c_-(x) \geq c_+(x)$. For any x, $c(x) = 1$, when both $c_+(x) = 1$ and $c_-(x) = 1$, $c(x) = 0$, when both $c_+(x) = 0$ and $c_-(x) = 0$ and "?" otherwise. A fully reliable classifier is a partial classifier which makes no errors. We define fully reliable learning as:

Full Reliable Learning. *Algorithm $\mathcal{A}$* efficiently fully reliably learns *a sequence of concept classes $\langle \mathcal{C}_n \rangle_{n\geq 1}$, under distributions $\langle \mathcal{D}_n \rangle_{n\geq 1}$, if there exists a polynomial $p(n, 1/\epsilon, 1/\delta)$ such that, for every $n \geq 1$, $\epsilon, \delta > 0$ and every $f_n : X_n \to [0, 1]$, algorithm $\mathcal{A}^{\mathcal{O}(f_n)}(\epsilon, \delta)$, with probability at least $1 - \delta$, outputs a hypothesis h such that* $\text{err}(h, \mathcal{D}_n, f_n) \leq \epsilon$ *and*

$$\text{acc}(h, \mathcal{D}_n, f_n) \geq \max_{c \in \mathsf{PC}(\mathcal{C}_n)} \text{acc}(c, \mathcal{D}_n, f_n) - \epsilon.$$

The time complexity of both $\mathcal{A}$ and h is bounded by $p(n, 1/\epsilon, 1/\delta)$.

In section 2 we gave an algorithm for positive reliable learning using a black-box agnostic learner for $\langle \mathcal{C} \rangle_{n\geq 1}$ (using Lemma 2). We define this as algorithm PRL (positive reliable learner). The input to the algorithm is ϵ - the accuracy parameter, δ - the confidence parameter and oracle $\mathcal{O}(f)$ - oracle access to the target function. The output of the algorithm is a hypothesis h such that $\text{false}_+(h) \leq \epsilon$ and $\text{false}_-(h) \leq \min_{c \in \mathcal{C}^+} \text{false}_-(c) + \epsilon$. The algorithm runs in time polynomial in n, $1/\epsilon$ and $1/\delta$. A symmetric algorithm NRL (negative reliable learner) also has identical properties with false positive and false negative error bounds interchanged. Our main result of this section is showing that agnostic learning implies fully reliable learning.

Theorem 3. *If $\mathcal{C}$ is efficiently agnostically learnable under distribution $\mathcal{D}$, it is efficiently fully reliably learnable under distribution $\mathcal{D}$.*

Proof. A simple algorithm to fully reliably learn $\mathcal{C}$ is the following: Let $h_+ = \mathsf{PRL}^{\mathcal{O}(f)}(\frac{\epsilon}{4}, \frac{\delta}{2})$, $h_- = \mathsf{NRL}^{\mathcal{O}(f)}(\frac{\epsilon}{4}, \frac{\delta}{2})$. Define $h : X \to \{0, 1, ?\}$ as,

$$h(x) = \begin{cases} 1 & \text{if } h_+(x) = h_-(x) = 1 \\ 0 & \text{if } h_+(x) = h_-(x) = 0 \\ ? & \text{otherwise} \end{cases}$$

We claim that hypothesis h is close to the best fully reliable partial classifier.

Let $c = (c_+, c_-) \in \mathsf{PC}(\mathcal{C})$ be such that $\text{err}(c, \mathcal{D}, f) = 0$ and $\text{acc}(c, \mathcal{D}, f) = \max_{c' \in \mathsf{PC}(\mathcal{C})} \text{acc}(c', \mathcal{D}, f)$. We know that the following hold:

$$\text{false}_+(h_+, \mathcal{D}, f) \leq \frac{\epsilon}{4}$$
$$\text{false}_-(h_+, \mathcal{D}, f) \leq \text{false}_-(c_+, \mathcal{D}, f) + \frac{\epsilon}{4}$$

Similarly,

$$\text{false}_-(h_-, \mathcal{D}, f) \leq \frac{\epsilon}{4}$$
$$\text{false}_+(h_-, \mathcal{D}, f) \leq \text{false}_+(c_-, \mathcal{D}, f) + \frac{\epsilon}{4}$$

Then $\text{err}(h, \mathcal{D}, f) \leq \text{false}_+(h_+, \mathcal{D}, f) + \text{false}_-(h_-, \mathcal{D}, f) \leq \frac{\epsilon}{2}$.

Note that when $h(x) =?$ then $h_+(x) \neq h_-(x)$ and hence one of them makes an error. Therefore, we also have $1 - \text{acc}(h, \mathcal{D}, f) \leq \text{err}(h_+, \mathcal{D}, f) + \text{err}(h_-, \mathcal{D}, f)$. We know that

$$\begin{aligned} \text{err}(h_+, \mathcal{D}, f) &= \text{false}_+(h_+, \mathcal{D}, f) + \text{false}_-(h_+, \mathcal{D}, f) \\ &\leq \text{false}_-(c_+, \mathcal{D}, f) + \frac{\epsilon}{2} \end{aligned}$$

Similarly $\text{err}(h_-, \mathcal{D}, f) \leq \text{false}_+(c_-, \mathcal{D}, f) + \epsilon/2$. Finally, we check that $\text{false}_-(c_+, \mathcal{D}, f) + \text{false}_+(c_-, \mathcal{D}, f) = 1 - \text{acc}(c, \mathcal{D}, f)$, hence $\text{acc}(h, \mathcal{D}, f) \geq \text{acc}(c, \mathcal{D}, f) - \epsilon$ □

4 Reliable learning with tolerance

In this section, we consider our final generalization where we allow tolerance rates τ_+ and τ_- for false positives and false negatives respectively. As in the case of fully reliable learning we need to consider the class of partial classifiers. Given a sequence of concept classes $\langle \mathcal{C}_n \rangle_{n\geq 1}$, for each $\mathcal{C}_n$ we define the class of sandwich classifiers as $\mathsf{SC}(\mathcal{C}_n) = \{(c_+, c_-) \mid c_+ \leq c_-\}$. For a sandwich classifier $c = (c_+, c_-)$, given x, $c(x) = 1$ when both $c_+(x) = 1$ and $c_-(x) = 1$, $c(x) = 0$ when both $c_+(x) = 0$ and $c_-(x) = 0$ and "?" otherwise.

Sandwich classifiers are a generalization of the class of partial classifiers $\mathsf{PC}(\mathcal{C}_n)$ that we defined in the previous section. In particular, the class $\mathsf{PC}(\mathcal{C}_n)$ is the set of all sandwich classifiers with zero error. We call a sandwich classifier $c \in \mathsf{SC}(\mathcal{C}_n)$, (τ_+, τ_-)-*tolerant reliable* if $\text{false}_+(c, \mathcal{D}_n, f_n) \leq \tau_+$ and $\text{false}_-(c, \mathcal{D}_n, f_n) \leq \tau_-$. Given acceptable tolerance rates of τ_+, τ_- define opt as

$$\begin{aligned} \text{opt}(\tau_+, \tau_-) \ &= \ \max_{c \in \mathsf{SC}(\mathcal{C}_n)} \text{acc}(c); \\ \text{subject to:} \quad &\{\text{false}_+(c) \leq \tau_+;\ \text{false}_-(c) \leq \tau_-\} \end{aligned}$$

We show that algorithm Sandwich Learner (Fig. 3) efficiently learns the class of halfspace sandwiches, over the unit ball $\mathcal{B}_n$, under the uniform distribution. With minor modifications, this algorithm can also learn under log-concave distributions. A halfspace sandwich over $\mathcal{B}_n$ can be visualized as two slices, one labeled 1, the other 0 and the remaining ball labeled "?". Our algorithm is based on the results due to Kalai, Klivans, Mansour and Servedio [KKMS05], and the analysis is largely similar. Algorithm Sandwich Learner in each iteration draws a sample of size m and sets up an ℓ_1 regression problem (8- 11), defined as:

$$\min_{\substack{\deg(p) \leq d \\ \deg(q) \leq d}} \frac{1}{m} \left(\sum_{i:\ y^i = 1} |1 - p(x^i)| + \sum_{i:\ y^i = 0} |q(x^i)| \right) \quad (8)$$

inputs: m, T, d
1.Draw m labeled examples
$Z^t = \langle (x^{t_1}, y^{t_1}), \ldots, (x^{t_m}, y^{t_m}) \rangle \in (\mathcal{B}_n \times \{0,1\})^m$
and set up the ℓ_1 polynomial regression problem (8-11).
2.Solve the regression problem, get polynomials $p^t(x), q^t(x)$ and record z^t to be the value of the objective function.
3.Repeat the above two steps T times and take $p = p^s, q = q^s$ to be the polynomials where z^s was the least.
4.Output a randomized partial hypothesis $h : \mathcal{B}_n \times \{0,1,?\} \to [0,1]$:

$$\begin{aligned} h(x,1) &= \mathrm{crop}(\min(p(x), q(x))) \\ h(x,0) &= \mathrm{crop}(\min(1-p(x), 1-q(x)) \\ h(x,?) &= 1 - h(x,1) - h(x,0) \end{aligned}$$

Figure 3: Algorithm Sandwich Learner

subject to

$$\frac{1}{m} \sum_{i:\ y^i=0} |p(x^i)| \leq \tau_+ + \frac{\epsilon}{2} \tag{9}$$

$$\frac{1}{m} \sum_{i:\ y^i=1} |1 - q(x^i)| \leq \tau_- + \frac{\epsilon}{2} \tag{10}$$

$$\frac{1}{m} \sum_i \max(p(x^i) - q(x^i), 0) \leq \frac{\epsilon}{8} \tag{11}$$

This problem can be solved efficiently by setting it up as a linear programming problem (see appendix of [KKMS05]). Thus, the running time of the algorithm is polynomial in m, T and d. The function $\mathrm{crop} : \mathbb{R} \to [0,1]$ used in the algorithm is defined as: $\mathrm{crop}(z) = z$ for $z \in [0,1]$, $\mathrm{crop}(z) = 0$ for $z < 0$ and $\mathrm{crop}(z) = 1$ for $z > 1$. The output of the algorithm is a randomized partial classifier. We define a *randomized partial classifier* as a function, $c : X \times \{0,1,?\} \to [0,1]$, so that $c(x,0) + c(x,1) + c(x,?) = 1$, and $c(x,y)$ is interpreted the probability that the output for x is y. We define error and empirical error of such classifiers as,

$$\mathrm{err}(c, \mathcal{D}) = \mathop{\mathbf{E}}_{(x,y)\sim\mathcal{D}}[c(x, 1-y)]$$

$$\widehat{\mathrm{err}}(c, Z) = \frac{1}{m} \sum_{i=1}^{m} c(x^i, 1-y^i),$$

Other quantities are defined similarly. Theorem 4 below shows that for any constant ϵ, the class of halfspace sandwiches over the unit ball $\mathcal{B}_n$ is efficiently (τ_+, τ_-)-tolerant reliably learnable. The running time of the algorithm is exponential in $1/\epsilon$.

Theorem 4. *Let $\mathcal{U}$ be the uniform distribution on the unit ball $\mathcal{B}_n$ and let $f : \mathcal{B}_n \to [0,1]$ be an arbitrary unknown function. There exists a polynomial P such that, for any $\epsilon, \delta > 0$, $n \geq 1$, $\tau_+, \tau_- \geq 0$, algorithm* Sandwich-Learner *with parameters $m = P(n^d/(\epsilon\delta))$, $T = \log(2/\delta)$, $d = O(1/\epsilon^2)$, with probability at least $1-\delta$, returns a randomized hypothesis h which satisfies $\mathrm{false}_+(h) \leq \tau_+ + \epsilon$, $\mathrm{false}_-(h) \leq \tau_- + \epsilon$ and $\mathrm{acc}(h) \geq \mathrm{opt}(\tau_+, \tau_-) - \epsilon$, where* opt *is with respect to the class of halfspace sandwich classifiers.*

The proof of Theorem 4 is given in the appendix.

References

[CHHS02] A. Cannon, J. Howse, D. Hush, and C. Scovel. Learning with the neyman-pearson and min-max criteria. Technical Report LA-UR-02-2951, Los Alamos National Laboratory, 2002.

[Elk01] C. Elkan. The foundations of cost-sensitive learning. In *Proceedings of the seventeenth International Joint Conference on Artificial Intelligence*, pages 973–978, 2001.

[FFHO04] C. Ferri, P. Flach, and J. Hernández-Orallo. Delegating classifier. In *ICML '04: Proceedings of the twenty-first International Conference on Machine Learning*, pages 37–44, New York, NY, USA, 2004. ACM.

[FHO04] C. Ferri and J. Hernández-Orallo. Cautious classifiers. In *Proceedings of first International Workshop on the ROC Analysis in Artificial Intelligence*, pages 27–36, 2004.

[GKK08] P. Gopalan, A.T. Kalai, and A.R. Klivans. Agnostically learning decision trees. In *STOC '08: Proceedings of the fortieth annual ACM Symposium on the Theory of Computation*, New York, NY, USA, 2008. ACM.

[Hau92] D. Haussler. Decision-theoretic generalizations of the PAC model for neural networks and other applications. *Information and Computation*, 100:78–150, 1992.

[Jac97] J.C. Jackson. An efficient membership-query algorithm for learning DNF with respect to the uniform distribution. *Journal of Computer and System Sciences*, 55(3):414–440, 1997.

[KKMS05] A.T. Kalai, A.R. Klivans, Y. Mansour, and R.A. Servedio. Agnostically learning halfspaces. In *FOCS '05: Proceedings of the 46th annual IEEE Symposium on Foundations of Computer Science*, pages 11–20, Washington, DC, USA, 2005. IEEE Computer Society.

[KSS94] M.J. Kearns, R.E. Schapire, and L.M. Sellie. Toward efficient agnostic learning. *Machine Learning*, 17(2):115–141, 1994.

[NP33] J. Neyman and E. S. Pearson. On the problem of the most efficient tests for statistical hypotheses. *Philosophical Transactions of the Royal Society of London, Series A, Containing Papers of a Mathematical or Physical Character*, 231:289–337, 1933.

[Pie07] Tadeusz Pietraszek. On the use of roc analysis for the optimization of abstaining clas-

sifiers. *Machine Learning*, 68(2):137–169, 2007.

[SN05] C. Scott and R. Nowak. A neyman-pearson approach to statistical learning. *IEEE Transactions on Information Theory*, 51(11):3806–3819, 2005.

[Val84] L.G. Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.

[ZLA03] B. Zadrozny, J. Langford, and N. Abe. Cost-sensitive learning by cost-proportionate example weighting. In *ICDM '03: Proceeding of the third IEEE International Conference on Data Mining*, page 435, Washington, DC, USA, 2003. IEEE Computer Society.

A Proof of Theorem 4

To prove Theorem 4, we need Lemma 8 as an intermediate step. Here $\mathcal{B}_n$ is the unit ball.

Lemma 8. *Let $\mathcal{D}$ be a distribution over $\mathcal{B}_n$ and $f : \mathcal{B}_n \to [0,1]$ be an arbitrary function. Let τ_+, τ_- be the required tolerance parameters. Suppose $c = (c_+, c_-)$ is a half-space sandwich classifier such that $\text{false}_+(c) \le \tau_+$, $\text{false}_-(c) \le \tau_-$ and $\text{acc}(c) = \text{opt}(\tau_+, \tau_-)$. If p_+, p_- are degree d polynomials such that $\mathbf{E}[|c_+(x) - p_+(x)|] \le \frac{\epsilon}{128}$ and $\mathbf{E}[|c_-(x) - p_-(x)|] \le \frac{\epsilon}{128}$, and if $m = \frac{128}{\epsilon^2}$, then with probability at least $\frac{1}{2}$, p_+, p_- are feasible solutions to the ℓ_1 Polynomial Regression Problem (8-11) and value of the objective function at p_+, p_- is at most $\text{false}_-(c_+) + \text{false}_+(c_-) + \frac{\epsilon}{2}$.*

Proof. Using Hoeffding's bound when $m = \frac{128}{\epsilon^2}$,

$$\Pr[\widehat{\text{false}}_+(c_+) \ge \text{false}_+(c_+) + \frac{\epsilon}{8}] \le \frac{1}{16} \qquad (12)$$

where $\widehat{\text{false}}_+(c_+)$ is the empirical estimate of $\text{false}_+(c_+)$ using a sample of size m. Also by Markov's inequality,

$$\Pr\left[\frac{1}{m}\sum_i |c_+(x^i) - p_+(x^i)| \ge \frac{\epsilon}{8}\right] \le$$
$$\Pr\left[\frac{1}{m}\sum_i |c_+(x^i) - p_+(x^i)| \ge 16\,\mathbf{E}[|c_+(x) - p_+(x)|]\right] \le \frac{1}{16} \qquad (13)$$

And hence with probability at least 7/8,

$$\begin{aligned}\frac{1}{m}\sum_{i:\ y^i=0} |p_+(x^i)| &\le \frac{1}{m}\sum_{i:\ y^i=0} (|c_+(x^i)| + |c_+(x^i) - p_+(x^i)|) \\ &\le \text{false}_+(c_+) + \epsilon/8 + \epsilon/8 \\ &\le \tau_+ + \epsilon/4 \qquad (14)\end{aligned}$$

and hence p_+ satisfies constraint (9) in the regression problem.

Similarly one can show that with probability at least 7/8, p_- satisfies constraint (10) of the regression problem.

Observe that $\mathbf{E}[\max(p_+(x) - p_-(x), 0)] \le \mathbf{E}[|p_+(x) - p_-(x) + c_-(x) - c_+(x)|] \le \mathbf{E}[|p_+(x) - c_+(x)|] + \mathbf{E}[|p_-(x) - c_-(x)|] \le \epsilon/64$. Here we use the fact that $c_-(x) \ge c_+(x)$ for all x (according to our definition of $\mathsf{SC}(\mathcal{C})$). Hence by Markov's inequality,

$$\Pr[\frac{1}{m}\sum_i \max(p_+(x^i) - p_i(x^i), 0) \ge \frac{\epsilon}{8}] \le \frac{1}{8} \qquad (15)$$

By union bound with probability at least 5/8, p_+, p_- satisfy all the constraints of the regression problem. Let us assume we are in the event where all of (12-15) and the corresponding statements in the case of false negatives are true. Using Hoeffding's bound, $\Pr[\widehat{\text{false}}_-(c_+) \ge \text{false}_-(c_+) + \epsilon/8] \le 1/16$ and $\Pr[\widehat{\text{false}}_+(c_-) \ge \text{false}_+(c_-) + \epsilon/8] \le 1/16$. We allow ourselves a further 1/8 loss in probability so that these two events do not occur either. Thus with probability at least 1/2, p_+, p_- are feasible solutions to the regression problem and the value of the objective is,

$$\begin{aligned}&\frac{1}{m}\sum_{i:\ y^i=1} |1 - p_+(x^i)| + \frac{1}{m}\sum_{i:\ y^i=0} |p_-(x^i)| \\ &\le \frac{1}{m}\sum_{i:\ y^i=1} (|1 - c_+(x^i)| + |c_+(x^i) - p_+(x^i)|) \\ &\quad + \frac{1}{m}\sum_{i:\ y^i=0} (|c_-(x^i)| + |c_-(x^i) - p_-(x^i)|) \\ &\le \text{false}_-(c_+) + \text{false}_+(c_-) + \epsilon/2\end{aligned}$$

□

Proof of Theorem 4. We use threshold functions θ_t in our proof, where for any $t \in [0,1]$, $\theta_t : \mathbb{R} \to \{0,1\}$ is such that $\theta_t(x) = 1$ if $x \ge t$, and $\theta_t(x) = 0$ otherwise. Although the threshold functions θ_t do not occur in the algorithm, they significantly simplify the proof. To get a decision using the randomized hypothesis that our algorithm outputs, a simple technique is to choose a threshold uniformly in $[0,1]$ and use that to output a value in $\{0, 1, ?\}$. Thresholds over polynomials are half-spaces in a higher (n^d) dimensional space, and we can use standard results from VC theory to bound the difference between the empirical error rates and true error rates. We will use frequently the following useful observation: For any $z \in \mathbb{R}$ we have

$$\text{crop}(z) = \int_0^1 \theta_t(z)dt.$$

For a degree d polynomial $p : \mathbb{R}^n \to \mathbb{R}$, the function $\theta_t \circ p$ can be viewed a half-space in dimension n^d dimension, where we extend the terms of the polynomial to a linear function. Using the classical VC theory, for any distribution $\mathcal{D}$ over $\mathcal{B}_n$, there exists a polynomial Q such that for $m = Q(n^d, \epsilon^{-1}, \delta^{-1})$ a sample S of m examples drawn from $(\mathcal{D}, f)$, with probability at least $1 - \delta$, $|\widehat{\text{err}}(\theta_t \circ p) - \text{err}(\theta_t \circ p)| \le \epsilon$, where $\widehat{\text{err}}(g) = \frac{1}{m}\sum_{(x,y)\in S} I(g(x) \ne y)$. Similar bounds hold for false_+ and false_-, where $\widehat{\text{false}}_+(g, S) = \frac{1}{m}\sum_{(x,y)\in S}(1 - g(x))y$ and $\widehat{\text{false}}_-(g, S) = \frac{1}{m}\sum_{(x,y)\in S} g(x)(1 - y)$.

Let $c = (c_+, c_-)$ be the best linear sandwich classifier with respect to tolerance rates τ_+, τ_-. Using results from Kalai, Klivans, Mansour and Servedio [KKMS05], for $d = O(1/\epsilon^2)$ there exist polynomials p_+, p_- such that $\mathbf{E}[|p_+(x) - c_+(x)|] \leq \epsilon/128$ and $\mathbf{E}[|p_-(x) - c_-(x)|] \leq \epsilon/128$. Thus when algorithm SANDWICH-LEARNER is run with $T = \log \frac{2}{\delta}$ by Lemma 8, with probability at least $1 - \delta/2$, at least one of the iterations has value of the objective function smaller than $\text{false}_-(c_+) + \text{false}_+(c_-) + \epsilon/2$. It can be checked that $\text{false}_-(c_+) + \text{false}_+(c_-) = 1 - \text{acc}(c) = \text{err}(c) + ?(c)$. We assume that we are in the case when the least objective function is smaller than $1 - \text{acc}(c) + \epsilon/2$, allowing our algorithm to fail with probability $\delta/2$ so far. Let p, q be the polynomials which are solutions to the regression problem with the least objective function.

We now analyze the quantities false_+, false_-, acc of the randomized hypothesis h output by algorithm SANDWICH-LEARNER. We present the analysis of false_+, and false_- can be done similarly. We assume that the number of examples m is large enough, so that all the bounds due to VC theory that we require in the proof hold simultaneously with probability at least $1 - \frac{\delta}{2}$.[1] This can be done easily by taking union bound using only polynomial number of examples, say $P(n^d/(\epsilon\delta))$.

For the first part, consider hyperplanes of the form $\theta_t \circ p$ and $\theta_t \circ (1-q)$, for any $t \in [0,1]$. By the VC bound we have that $|\widehat{\text{false}}_+(\theta_t \circ p, S) - \text{false}_+(\theta_t \circ p, \mathcal{U}, f)| \leq \epsilon/2$ and $|\widehat{\text{false}}_-(\theta_t \circ (1-q), S) - \text{false}_+(\theta_t \circ (1-q), \mathcal{U}, f)| \leq \epsilon/2$. The analysis then proceeds as follows,

$$\begin{aligned}
\text{false}_+(h, \mathcal{U}, f) &= \mathop{\mathbf{E}}_{x \sim \mathcal{U}}[h(x,1)(1 - f(x))] \\
&= \mathop{\mathbf{E}}_{x \sim \mathcal{U}}[\int_0^1 \theta_t(\min(p(x), q(x)))dt(1 - f(x))] \\
&\leq \int_0^1 \mathop{\mathbf{E}}_{x \sim \mathcal{U}}[\theta_t(p(x))(1 - f(x))]dt \\
&= \int_0^1 \text{false}_+(\theta_t \circ p, \mathcal{U}, f)dt \\
&\leq \int_0^1 \widehat{\text{false}}_+(\theta_t \circ p, S)dt + \frac{\epsilon}{2} \\
&= \int_0^1 \frac{1}{m} \sum_{i:\ y^i = 0} \theta_t(p(x^i))dt + \frac{\epsilon}{2} \\
&\leq \frac{1}{m} \sum_{i:\ y^i = 0} |p(x^i)| + \frac{\epsilon}{2} \\
&\leq \tau_+ + \epsilon
\end{aligned}$$

Next we analyze the quantity $1 - \text{acc}(h) = \text{err}(h) + ?(h)$. Let $\bar{S} = \{(x^1, 1 - y^1), \ldots, (x^m, 1 - y^m)\}$, namely we flip the label in S, and $S^0 = \{(x^1, 0), \ldots, (x^m, 0)\}$, namely the sample S all with labels 0. Here we assume the following bounds hold $|\widehat{\text{false}}_+(\theta_t \circ (1-p), \bar{S}) - \text{false}_+(\theta_t \circ (1-p), \mathcal{U}, 1-f)| \leq \epsilon/8$, $|\widehat{\text{false}}_+(\theta_t \circ q, S) - \text{false}_+(\theta_t \circ q, \mathcal{U}, f)| \leq \epsilon/8$ and $|\widehat{\text{err}}(\theta_t \circ (p-q), S^0) - \text{err}(\theta_t \circ (p-q), \mathcal{U}, 0)| \leq \epsilon/8$. Step (16) below holds because $1 - \text{crop}(a) = \text{crop}(1-a)$. Step (17) uses the fact that $\min(a,b) = a - (a-b)\text{I}(a > b)$ and $\max(a,b) = b + (a-b)\text{I}(a > b)$. Finally in step (18) we use that $\text{crop}(a+b) \leq \text{crop}(a) + \text{crop}(b)$. All these facts can be checked easily.

$$\begin{aligned}
&1 - \text{acc}(h) \\
&= 1 - \mathop{\mathbf{E}}_{x \in \mathcal{U}}[h(x,1)f(x) + h(x,0)(1 - f(x))] \\
&= \mathop{\mathbf{E}}_{x \in \mathcal{U}}[(1 - h(x,1))f(x) + (1 - h(x,0))(1 - f(x))] \\
&= \mathop{\mathbf{E}}_{x \in \mathcal{U}}[(1 - \text{crop}(\min(p(x), q(x))))f(x) + \\
&\quad (1 - \text{crop}(\min(1 - p(x), 1 - q(x))))(1 - f(x))] \\
&= \mathop{\mathbf{E}}_{x \in \mathcal{U}}[\text{crop}(1 - \min(p(x), q(x)))f(x) + \\
&\quad \text{crop}(\max(p(x), q(x)))(1 - f(x))] \qquad (16) \\
&= \mathop{\mathbf{E}}_{x \in \mathcal{U}}[\text{crop}(1 - p(x) + (p(x) - q(x))\text{I}(p(x) > q(x)))f(x) \\
&\qquad + \text{crop}(q(x) + (p(x) - q(x))\text{I}(p(x) > q(x)))(1 - f(x))] \\
&\qquad\qquad (17) \\
&\leq \mathop{\mathbf{E}}_{x \in \mathcal{U}}[\text{crop}(1 - p(x))f(x) + \text{crop}(q(x))(1 - f(x)) \\
&\quad + \text{crop}(p(x) - q(x))] \qquad (18) \\
&= \mathop{\mathbf{E}}_{x \in \mathcal{U}}[\int_0^1 \theta_t(1 - p(x))f(x)dt + \int_0^1 \theta_t q(x)(1 - f(x))dt \\
&\quad + \int_0^1 \theta_t(p(x) - q(x)) \cdot 1 \ \ dt] \\
&= \int_0^1 \left(\mathop{\mathbf{E}}_{x \in \mathcal{U}}[\theta_t(1 - p(x))f(x)] + \mathop{\mathbf{E}}_{x \in \mathcal{U}}[\theta_t(q(x))(1 - f(x))] \right. \\
&\quad \left. + \mathop{\mathbf{E}}_{x \in \mathcal{U}}[\theta_t(p(x) - q(x))] \right) dt \\
&= \int_0^1 (\text{false}_+(\theta_t \circ (1-p), \mathcal{U}, 1 - f) + \text{false}_+(\theta_t \circ q, \mathcal{U}, f) \\
&\quad + \text{err}(\theta_t \circ (p - q), \mathcal{U}, 0))\, dt \\
&\leq \int_0^1 \left(\widehat{\text{false}}_+(\theta_t \circ (1-p), \bar{S}) + \widehat{\text{false}}_+(\theta_t \circ q, S) \right. \\
&\quad \left. + \widehat{\text{err}}(\theta_t \circ (p - q), S^0) \right) dt + \frac{3\epsilon}{8} \\
&= \int_0^1 \left(\frac{1}{m} \sum_{i:\ y^i = 1} \theta_t(1 - p(x^i)) + \frac{1}{m} \sum_{i:\ y^i = 0} \theta_t(q(x^i)) \right. \\
&\quad \left. + \frac{1}{m} \sum_{i=1}^m \theta_t(p(x^i) - q(x^i)) \right) dt + \frac{3\epsilon}{8} \\
&= \frac{1}{m} \sum_{i:\ y^i = 1} \int_0^1 \theta_t(1 - p(x^i))dt + \frac{1}{m} \sum_{i:\ y^i = 0} \int_0^1 \theta_t(q(x^i))dt \\
&\quad + \frac{1}{m} \sum_i \int_0^1 \theta_t(p(x^i) - q(x^i))dt + \frac{3\epsilon}{8} \\
&\qquad\qquad (19)
\end{aligned}$$

[1] All our requirements would be about comparing the error of a hyperplane to its observed error on the sample. We would need that the sample $S = \{(x^1, y^1), \ldots, (x^m, y^m)\}$ is such that no hyperplane would have a difference larger than $\epsilon/8$.

$$
\begin{aligned}
&\leq \frac{1}{m} \sum_{i:\ y^i=1} |1 - p(x^i)| + \frac{1}{m} \sum_{i:\ y^i=0} |q(x^i)| \\
&\quad + \frac{1}{m} \sum_{i=1}^{m} \max(p(x^i) - q(x^i), 0) + \frac{3\epsilon}{8} \\
&\leq 1 - \mathrm{acc}(c, \mathcal{U}, f) + \epsilon
\end{aligned}
$$

where in the last inequality we used that fact that the first two terms are the objective function of the regression (and we assumed that they are at most $\mathrm{false}_-(c_+) + \mathrm{false}_+(c_-) + \epsilon/2$, and the third term is bounded in the regression by $\epsilon/8$. Hence $\mathrm{acc}(h) \geq \mathrm{acc}(c) - \epsilon$. $\square$