Learning Fixed-dimension Linear Thresholds From Fragmented Data*

Paul W. Goldberg
Dept. of Computer Science,
University of Warwick,
Coventry CV4 7AL, U.K.
pwg@dcs.warwick.ac.uk

January 3, 2001

Abstract

We investigate PAC-learning in a situation in which examples (consisting of an input vector and 0/1 label) have some of the components of the input vector concealed from the learner. This is a special case of Restricted Focus of Attention (RFA) learning. Our interest here is in 1-RFA learning, where only a single component of an input vector is given, for each example. We argue that 1-RFA learning merits special consideration within the wider field of RFA learning. It is the most restrictive form of RFA learning (so that positive results apply in general), and it models a type of "data fusion" scenario, where we have sets of observations from a number of separate sensors, but these sensors are uncorrelated sources.

Within this setting we study the well-known class of linear threshold functions, the characteristic functions of Euclidean half-spaces. The sample complexity (*i.e.* sample-size requirement as a function of the parameters) of this learning problem is affected by the input distribution. We show that the sample complexity is always finite, for any given input distribution, but we also exhibit methods for defining "bad" input distributions for which the sample complexity can grow arbitrarily fast. We identify fairly general sufficient conditions for an input distribution to give rise to sample complexity that is polynomial in the PAC parameters ϵ^{-1} and δ^{-1}. We give an algorithm whose sample complexity is polynomial in these parameters and the dimension (number of input components), for input distributions that satisfy our conditions. The runtime is polynomial in ϵ^{-1} and δ^{-1} provided that the dimension is any constant. We show how to adapt the algorithm to handle uniform misclassification noise.

*A preliminary version of this paper appeared in the proceedings of the 1999 COLT conference.

1 Introduction

The aim of *supervised learning* is to find out as much as possible about some unknown function (called the *target function*) using observations of its input/output behavior. In this paper we focus on linear threshold functions. These functions map vectors of inputs to binary outputs according to the rule that the output should equal 1 provided that some linear combination of the inputs exceeds some threshold value, otherwise the output equals 0. Thus a linear threshold function can be described by a vector of real coefficients, one for each input, and a real-valued threshold.

Probably Approximately Correct (PAC) learning is a well-known framework for studying supervised learning problems in which outputs of the functions under consideration may take one of two values (such as 0 and 1), so that any function partitions the input domain into two sets. We give the basic definitions of PAC learning below in section 1.2; see textbooks such as [2, 31] for a detailed introduction to the theory.

The problem of learning linear threshold functions in the PAC framework has received a lot of attention in the literature, some of which is described below. In this paper we consider a natural variant of the problem in which the algorithm has access to examples of the target function in which only a single input component value (together with the output value, 0 or 1) is given. It is assumed that for each example of input/output behavior, the choice of which component has its value given, is made uniformly at random.

The paper is organized as follows. In this section we give background, motivation for studying this variant in detail, a formal statement of the learning situation, and some preliminary results. In section 2 we show how the joint distribution of the inputs may affect the number of examples needed to distinguish the target function from a single alternative linear threshold function, having some given error. In section 3 we use a general method identified in section 2 to PAC-learn linear threshold functions, for any constant number of inputs. In section 4 we consider the special case where inputs are binary-valued. In section 5 we discuss the significance of the results presented here, and mention open problems of particular interest.

1.1 Background and Motivation

The topic of *missing data*, where some of the components of an observation are concealed from the learner, has received a lot of attention in the statistics literature. Within PAC learning theory the situation is called Restricted Focus of Attention (RFA) learning, introduced in [5, 6, 8], see [20] for an extensive survey. For query-based learning the associated framework is the Unspecified Attribute Values learning of [24]. A good example of a data set that motivates the work here is a medical prognosis problem analysed in Titterington et al. [37] and Lowe and Webb [33]. The data set represents 1000 head-injured coma patients, and contains (for each patient) a subset of a set of 6 diagnostic indicators measured on admission to hospital, and a measure of extent of recovery. The aim is to use the data to learn to predict recovery given new sets of measurements. In the data set, fewer than half of the patients had all 6 measurements taken, so there is a problem of how to use the

incomplete vectors of observations effectively.

Most methods for learning from incomplete data use *imputation*, in which the missing values in the data set are assigned values according to some rule (for example [33] use *mean imputation*, where an unknown component value is given the average of the known values for that component). In general, imputation biases the data slightly, which is at odds with the PAC criterion for successful learning, being used here. Linear threshold functions are an oversimplified model for the data, since there is class overlap (indeed the data set contains identical pairs of input vectors with distinct recovery levels). However our algorithm is extendable to a more realistic "misclassification noise" model.

Our simplifying assumption that each example has only a single input attribute value given has the following motivations:

1. It eliminates the strategy of discarding incomplete examples, which is wasteful in practice. The strategy of discarding incomplete examples may also bias the data if the missing data mechanism is more likely to conceal some values than others (*i.e.* anything other than what Little and Rubin [32] call *missing completely at random*).

2. The restriction to a constant number of values per example is equivalent to a simple stochastic missing-data mechanism, as well as being a special case of RFA learning. The statistical missing data literature usually assumes that there is a stochastic missing data mechanism, as opposed to RFA learning where unconcealed values are selected by the learner.

   k-RFA learning refers to a setting where k components of any example are known to the learner; thus we focus on 1-RFA learning. The equivalence noted above can be seen by observing that in our setting a learner may gather polynomial-sized collections of samples for each set of k attributes, as easily as it may gather a polynomial-sized sample, and hence effectively query any given set of k attributes. We prefer the term "fragmented data" over "missing data" in this situation, to emphasise that only a small proportion of any data vector is given.

3. The 1-RFA setting is the most stringent or restrictive situation, in that positive results for 1-RFA learning apply in other settings. It also models a "data fusion" problem, in which collections of examples are generated by a set of independent sources, and the aim is to combine (or "fuse") the information derived from the separate sources.

Linear threshold functions are an obvious choice of function class in the context introduced here, because the output value generally depends on all the input values; it is not generally sufficient to know just a subset of them. But information is still conveyed by an example in which all but one input value is concealed.

We next motivate the study of *distribution-specific* learning in this missing-data setting. This is justified mainly by the results, which show that the learning problem is impossible in a completely distribution-free setting (fact 1 below) and that the sample complexity depends on the input distribution (section 2). There has been relevant work on distribution-specific PAC learning in the standard complete data setting, see section 1.3. Work in RFA

learning generally assumes that the input distribution belongs to some known class, such as product distributions. It is known from this work that it is necessary to already have a lot of knowledge of the input distribution, in order to learn the function. We might reasonably expect to have a parametric model for the input distribution, and then use the EM algorithm [19] or subsequent related methods that have been devised for learning a distribution in the presence of missing data.

In section 2 we focus on the question of which distributions are helpful or unhelpful for 1-RFA learning. The sensitivity of the sample complexity to the nature of the input distribution (particularly when we do not restrict to product distributions) is a distinctive novel feature of this computational learning problem, with a lot of theoretical interest. (By sample complexity we mean the number of examples needed for PAC learning by a computationally unbounded learner.) Experimental work in the data fusion literature such as [12, 18] has shown the strong effect that varying assumptions about the input distribution may have on predictive performance. We aim to provide some theoretical explanation by identifying features of an input distribution that make it "helpful" and give associated sample-size bounds.

We mention relationships with other learning frameworks. The RFA setting is more benign than the "random attribute noise" [25, 36] scenario. A data set with missing components can be converted to one with random attribute noise by inserting random values for the missing components (although note that for k-RFA data, with small k, the associated noise rate would be quite high).

Finally, observe that there is a similarity to the *probabilistic concepts* framework of [30] in that, given a stochastic missing data mechanism, we have observations of a mapping from an input domain consisting of partially observed vectors to outputs whose values are conditional distributions over $\{0,1\}$ conditioned on the observed inputs. The difference is that we do not just want to model the conditional distribution of outputs given any input, we also want an underlying deterministic function to be well-approximated by our (deterministic) hypothesis. In this paper we make use of the *quadratic loss* function of an observation and hypothesis, as defined in [30].

1.2 Formalization of the Learning Problem

We are interested in algorithms for probably approximately correct (PAC) learning as introduced by Valiant in [38, 39]. Here we give the basic definitions and introduce some notation. An algorithm has access to a source of observations of a target function $F : X \to \{0,1\}$, in which inputs are chosen according to some fixed probability distribution D over the domain X, and the correct 0/1 output is given for each input. It is given two parameters, a target accuracy ϵ and an uncertainty bound δ. The goal is to output (in time polynomial in ϵ^{-1} and δ^{-1}), with probability at least $1-\delta$, a function $H : X \to \{0,1\}$ with the property that for random input chosen according to D, the probability that the output of H disagrees with the output of F, is at most ϵ. The input distribution D is usually assumed to be unknown, but the target function is known to belong to some given class $\mathcal{C}$ of functions.

Unlike most work on PAC learning, we assume that D is known completely (as studied in [7]). The RFA literature gives examples that show that some knowledge of D is necessary for most learning problems, and it is often assumed that D is a product distribution (each attribute chosen independently). In this paper we do not address the topic of partial knowledge of D. In the next section we show that some knowledge is necessary for learning linear threshold functions (the function class of interest here).

Within the PAC framework, we are studying specifically 1-RFA learnability where for each example the learner can see one of the input values and the binary output value. Thus, for domain $X = \mathbf{R}^d$, an example is a member of $\mathbf{R} \times \{1, \ldots, d\} \times \{0, 1\}$, since it contains a real value, the identity of the coordinate taking that value, and the output label. As noted, the assumption that the coordinate's identity is chosen by the learner is equivalent (for PAC learning) to the assumption that it is chosen at random. This is more stringent than "missing completely at random" since we have imposed an artificial limit (of 1) on the number of observed input values. We have observed that this artificial limit is important to disallow discarding some training examples and using others. Obviously PAC-learnability of 1-RFA data implies PAC-learnability of k-RFA data for any larger k.

Our aim is to use fragmented data to learn linear threshold functions, that is functions mapping members of some unknown halfspace of $\mathbf{R}^d$ to the output 0, and its complement to 1. These are functions of the form $f((x_1, \ldots, x_d)) = 1$ iff $\sum_i a_i x_i > \tau$ where a_i are unknown coefficients and τ is a "threshold" value. Throughout, we use the unit cost model of real number representation.

Our algorithm is (for a large class of input distributions) polynomial in the PAC parameters ϵ^{-1} and δ^{-1}, provided that d is constant. In investigating the behavior of the algorithm as a function of dimension d, we need to consider it with respect to a paramaterized class D_d of input distributions, where D_d is a probability distribution over $\mathbf{R}^d$. (This is due to the dependence we have noted of sample complexity on input distribution.) The algorithm's runtime is typically exponential in d, but for two classes D_d of interest, the sample complexity can be shown to be polynomial.

1.3 Related Work on Linear Thresholds and Noise-tolerant Learning

The domain $\mathbf{R}^d$ (for constant d) is a widely considered domain in the learning theory literature. Examples of learning problems over this domain include PAC-learning of boolean combinations of halfspaces [16], query-based learning of unions of boxes [15], and unions of halfspaces [11, 4, 13]. A technique of [11] generalized by [16] involves generating a set of functions that realise all linear partitions of a sample of input vectors. If m is the sample size then the set of partitions has size $O(m^d)$. Our algorithm uses this technique, which requires d to be constant for polynomial runtime. Extending the above learning results to general (non constant) d would solve the well-known open problem of learning disjunctive normal form boolean formulae, introduced in [39]. We explain below why it is likely to be difficult to generalize the results here to non-constant d.

Linear threshold functions have been studied extensively in the machine learning literature. We will not review the algorithms here, but see Blum et al. [10] for a good account of the PAC learning results. It is well-known that in the basic PAC framework, linear threshold functions are learnable. Finding a consistent hypothesis (a hyperplane that separates the given inputs with output 1 from those with output 0) can be solved by linear programming in polynomial time. The well-known results of Blumer et al. [11] show that any consistent hypothesis achieves PAC-ness, given a sample whose size is proportional to ϵ^{-1}, $\log(\delta^{-1})$, and d. (This uses the fact that the Vapnik-Chervonenkis (V-C) dimension of halfspaces of $\mathbf{R}^d$ is $d+1$, see [11] for details.)

As mentioned in the previous subsection, we assume unit cost for representation and arithmetic operations on real values. The algorithm of [10] PAC-learns linear threshold functions in the presence of random misclassification noise, and requires the logarithmic cost model for real value representation. So also does the basic PAC algorithm of [11], since known algorithms for linear programming that are polynomial in d assume logarithmic cost. (For unit cost real arithmetic, currently it is known how to do linear programming in polynomial time for logarithmic d, see Gärtner and Welzl [23].) These observations raise the question of whether we can find an algorithm that is polynomial in d as well as the PAC parameters, for logarithmic cost real arithmetic. In section 4 where we discuss in more detail the case where inputs come from the discrete boolean domain, we explain why this open problem is still likely to be hard.

In this paper we show how to convert our algorithm into a statistical query (SQ) algorithm (as introduced by Kearns [28]), which implies that it can be made noise-tolerant. (Over the boolean domain $\{0,1\}^d$ a more general result of this kind already exists, namely that learnability in the k-RFA setting implies SQ-learnability and hence learnability in the presence of random classification noise, for k logarithmic in d [6].) An extension to RFA learnability of linear thresholds (in time polynomial in d) would then be a strengthening of the result of [10].

Note that if we had a method for determining a good approximation of the error of a hypothesis (using the fragmented data) then we could PAC-learn, using a result of [7], which says that PAC-learnability with a known distribution D in the standard setting is equivalent to PAC-learnability with a known distribution when instead of examples, the learning algorithm has a means of measuring the error of any hypothesis it chooses. However, we have not found any general way of approximately measuring misclassification rate of a hypothesis using RFA data, even for the kinds of input distributions that we identify as implying polynomial sample complexity.

1.4 Technical Preliminaries

We establish some simple facts about the learning situation under consideration. These are to justify our assumption that the input distribution is not completely unknown. Note that learning may still be possible if the input distribution is not known completely, but known to belong to a class of distributions. In previous work on RFA learning, it is assumed that the input distribution D is an unknown *product* distribution. This is a strong assumption

which allows RFA data to convey a lot of information about D. It is already known from [5] that without some information about the input distribution it is often possible to define pairs of *scenarios* (a scenario is the combination of an input distribution and classifier) which are substantially different but are indistinguishable to a RFA learner. We use the same method for linear threshold functions.

Given a binary-valued function F, define $pos(F)$ to be the positive examples of F, *i.e.* $\{x : F(x) = 1\}$ and $neg(F)$ to be the negative examples, *i.e.* $\{x : F(x) = 0\}$.

Fact 1 *It is impossible to learn linear thresholds over* $\mathbf{R}^2$ *for a completely unknown input distribution* D*, even for a computationally unbounded learner.*

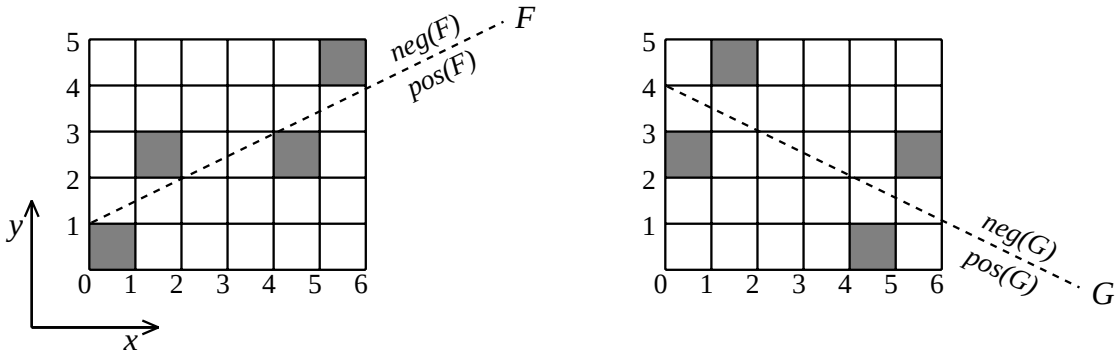

figure 1

Different but indistinguishable scenarios described in proof of fact 1.

Proof: Define linear threshold functions F, G over the (x, y)-plane as follows.

$$\begin{aligned} pos(F) = &\ \{(x, y) : y < 1 + x/2\} \\ pos(G) = &\ \{(x, y) : y < 4 - x/2\} \end{aligned}$$

Define input distributions D, D' over the (x, y)-plane as follows. D is uniform over the 4 unit squares whose lower-left corners are at $(0, 0)$, $(4, 2)$, $(1, 2)$ and $(5, 4)$. D' is uniform over the 4 unit squares with lower-left corners at $(0, 2)$, $(4, 0)$, $(1, 4)$ and $(5, 2)$. (These are the shaded regions in figure 1.)

Consider 1-RFA data generated by either F in combination with D, or G in combination with D'. The marginal distributions (that is, the distributions of the separate x and y coordinates) are the same in both cases, as are the conditional distributions of the output label given the input (so for example, $Pr(label = 1 \mid x \in [0, 1]) = 1$ in both cases, or $Pr(label = 1 \mid y \in [2, 3]) = 1/2$ in both cases). But the two underlying functions are very different. □

The discrete boolean domain $X = \{0,1\}^d$ is of special interest, and in section 4 we note the existence of a similar construction for that special case, thus showing that some knowledge of D is still required. (That construction uses 3 input dimensions, rather than just 2.)

The above construction gives indistinguishable scenarios for pairs of input distributions that differ from each other. We show later that for any known input distribution, there are no indistinguishable pairs of linear threshold functions (in contrast with function classes containing, for example, exclusive-or and its negation, [5]). But the following example shows how a known input distribution may affect sample complexity. Observe first that for pairwise comparison, the optimal strategy is to maximize the likelihood of the output labels given the input coordinate values. For an individual example in which the input coordinate x_i takes the value $r \in \mathbf{R}$ and the output label is $l \in \{0,1\}$, this likelihood is the probability that points generated by D conditioned on $x_i = r$ give output value l. For a collection of such examples the likelihood is the product of the individual likelihoods.

Example 2 *Suppose that D is uniform over two line segments in the (x,y)-plane, having (for some small positive ξ) endpoints $((\xi,0),(1,1-\xi))$ and $((0,\xi),(1-\xi,1))$. Let $F(x,y) = 1$ iff $y < x$ and let $G(x,y) = 1$ iff $y > x$.*

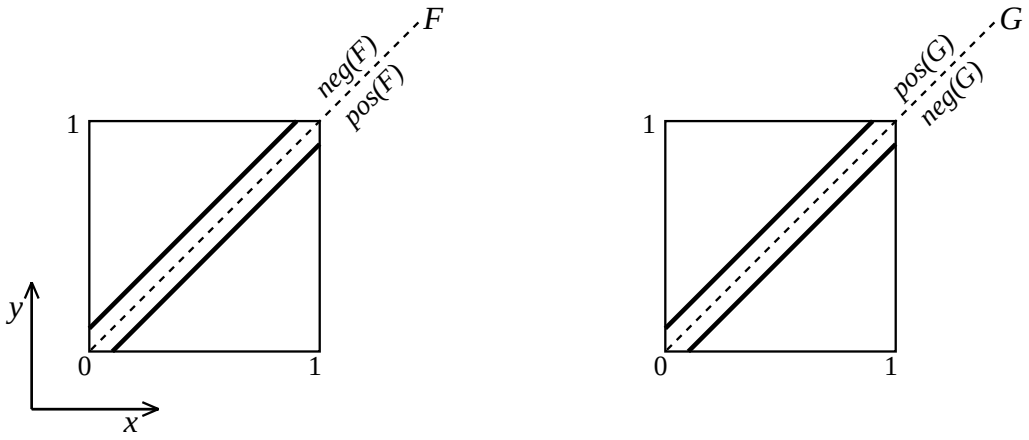

figure 2

F and G as defined in example 2, which disagree on all inputs (x,y). D is uniform over the two heavy line segments in the square.

If the target function is F (respectively, G), then a PAC algorithm should have a probability $\leq \delta$ of outputting G (respectively, F), for any error bound $\epsilon < 1$. But if either F or G is the target function, then in order to have any evidence in favor of one over the other, it is necessary to see an example in which the value of the given input coordinate lies in the range $[0,\xi) \cup (1-\xi,1]$. Examples of this kind occur with probability 2ξ, and all other points are uninformative (having equal likelihood for F and G). So the sample size needed

for PAC-learning is proportional to $1/\xi$, for this particular kind of input distribution. Note however that if we put $\xi = 0$ (and the domain becomes the line segment with endpoints at $(0,0)$ and $(1,1)$), the associated sample-size requirements do not become infinite; instead the learning problem reduces to a similar one in one dimension fewer.

2 Effect of Joint Distribution of Inputs on Sample Complexity of Pairwise Comparisons

In this section we give results about the way the joint distribution over input components may affect the sample-size requirements for a restriction of the learning problem. We assume that only two candidate functions F, G are given, which disagree with probability ϵ. One of them is the target function, and the aim is to determine which one is the target function, with probability $1 - \delta$ of correctness. Example 2 showed a class of input distributions whose members could make arbitrarily large the expected number of examples needed to distinguish a particular pair of functions. Note, however, that

1. No input distribution gave the requirement that any pair of positive values (ϵ, δ) of target accuracy and confidence required infinite data.

2. The *asymptotic* behaviour of sample-size requirements is still polynomial. In particular, we claim that given any pair of linear threshold functions that disagree with probability ϵ, we need $\Theta(\max(\epsilon^{-1}, \xi^{-1}))$ examples in order to distinguish them with some given probability of success. This is still polynomial in ϵ, for any given $\xi > 0$.

Regarding point 1 above, we show in section 2.1 (theorem 4) that there is no input distribution whose marginal distributions have well-defined means and variances that allows some pair of distinct linear threshold functions that differ by some $\epsilon > 0$ to be indistinguishable in the limit of infinite 1-RFA data. Moreover in corollary 5 we show that a finite upper bound on sample size can be derived from D, ϵ and δ only, and not on the particular choice of F and G which differ by ϵ. Regarding point 2, in section 2.2 we give fairly general sufficient conditions on an input distribution, for sample complexity to be polynomial. We do however in section 2.3 identify certain "pathological" distributions where the sample complexity is not necessarily polynomial in ϵ^{-1} and δ^{-1}.

2.1 Finiteness Results for Sample-size Requirements

In what follows, we assume that all probability distributions have well-defined expectations and variances for components of input vectors. Regarding point 1 above, we show that for these probability distributions there is never an infinite sample-size requirement once a distribution is given, despite the fact that distributions may be arbitrarily bad.

Lemma 3 *Let D, D' be probability distributions with domains R and R' respectively, both subsets of $\mathbf{R}^d$. Suppose moreover that R and R' are convex and do not intersect. Then for*

random variables x and x' generated by D and D' respectively, the expected values $E(x)$ and $E(x')$ are distinct.

Proof: Since the expected value is a convex combination, we just note that $E(x) \in R$ and $E(x') \in R'$, and since $R \cap R' = \emptyset$, the expected values are indeed distinct. □

F and G as defined in the statement of the following theorem are slightly more general than linear threshold functions — we use the additional generality in the proof of corollary 5. Recall that for any function $f : X \longrightarrow \{0,1\}$, $pos(f)$ denotes $\{x \in X \ : \ f(x) = 1\}$ and $neg(f)$ denotes $\{x \in X \ : \ f(x) = 0\}$.

Theorem 4 *Let D be any probability distribution over $\mathbf{R}^d$ whose marginal distributions have well-defined means and variances. Let F and G be any pair of functions from $\mathbf{R}^d$ to $\{0,1\}$ such that*

1. *$pos(F)$, $neg(F)$, $pos(G)$, $neg(G)$ are all convex.*
2. *with probability 1, a point generated by D lies in $pos(F) \cup neg(F)$.*
3. *with probability 1, a point generated by D lies in $pos(G) \cup neg(G)$.*
4. *with probability ϵ, a point generated by D is given different labels by F and G.*

Then F and G are distinguishable (using 1-RFA data) with probability $1 - \delta$ (for $\epsilon, \delta > 0$) for some sufficiently large finite sample size (dependent on $D, \epsilon, \delta, F, G$).

Proof: F and G divide the domain $\mathbf{R}^d$ into 4 convex regions defined as follows.

$$R_{00} = neg(F) \cap neg(G) \qquad R_{01} = neg(F) \cap pos(G)$$
$$R_{10} = pos(F) \cap neg(G) \qquad R_{11} = pos(F) \cap pos(G)$$

For $R \subseteq \mathbf{R}^d$ (where R is either one or a union of two of the R_{ij} regions) let $D(R)$ be the probability that a point generated by D lies in region R. The region of disagreement of F and G is $R_{01} \cup R_{10}$ – by assumption 4, $D(R_{01} \cup R_{10}) = \epsilon$. Let $\mu(R)$ denote the expectation of points generated by D, restricted to the region R — as long as $D(R) > 0$, $\mu(R)$ is well-defined by our assumption that components of points generated by D have well-defined expectations and variances.

We first consider the case that $D(R_{ij}) > 0$ for all $i, j \in \{0,1\}$.

The points $\mu(R_{00}), \mu(R_{01}), \mu(R_{10}), \mu(R_{11})$ are all distinct from each other (observing that the R_{ij} are convex and disjoint, so we can use lemma 3). Next note that the expected value of negative examples of F is a weighted average of $\mu(R_{00})$ and $\mu(R_{01})$ (weighted by probabilities $D(R_{00})$ and $D(R_{01})$). Similarly the expected value of negative examples of G is a weighted average of $\mu(R_{00})$ and $\mu(R_{10})$ (weighted by probabilities $D(R_{00})$ and $D(R_{10})$).

We use the fact $D(R_{01}) + D(R_{10}) = \epsilon > 0$ to deduce that the negative examples of F and G have different expectations. If the (distinct) points $\mu(R_{00})$, $\mu(R_{01})$, $\mu(R_{10})$ do not

lie on a one-dimensional line, this follows. If they lie on a line, the point $\mu(R_{01})$ cannot be in the middle, since that would contradict convexity of $neg(G)$. Similarly $\mu(R_{10})$ cannot lie in the middle. If $\mu(R_{00})$ lies between the other two, then observe that since the weights of the averages are positive, the means $\mu(neg(F))$ and $\mu(neg(G))$ must lie on opposite sides of $\mu(R_{00})$ on the line.

So we can choose a component on which means of negative examples differ, and use the observed mean of 0-labeled observations of that component to estimate the true expected value. Given our assumption that the variance is well-defined (finite), there will be a sufficiently large sample size such that we can with high probability predict which of F or G is labeling the data.

Now suppose that not all values $D(R_{ij})$ are positive. If R_{01} (respectively R_{10}) has zero probability measure, then $D(R_{10}) = \epsilon$ (respectively $D(R_{01}) = \epsilon$) and F and G can be distinguished using the relative frequencies of positive/negative examples. If we have $D(R_{00}) = 0$ then the argument for the "all positive" case goes through, since we assign weight zero to the undefined value $\mu(R_{00})$. □

Corollary 5 *Given any input distribution D over $\mathbf{R}^d$ and any target values $\epsilon, \delta > 0$ of PAC parameters, there exists a sufficiently large finite sample size for which* any *pair F, G of linear threshold functions can be distinguished with probability $1 - \delta$.*

Proof: Suppose otherwise. Then for some D, ϵ, δ there would exist a sequence of pairs $\{(F_i, G_i),\ i \in \mathbf{N}\}$ where F_i differs from G_i by ϵ, and as i increases, the sample-size required to distinguish F_i from G_i increases monotonically without limit. We prove by contradiction that such a sequence cannot exist.

The general strategy is as follows. From the sequence $\{(F_i, G_i)\}$ extract a subsequence $\{(F_i'', G_i'')\}$ which "converges" in the sense that as i increases, the probability of disagreement between F_i'' and F_j'', for any $j > i$, tends to zero, and likewise for G_i'' and G_j''. The sequences $\{F_i''\}$ and $\{G_i''\}$ then converge pointwise to binary classifiers F_∞ and G_∞ such that $pos(F_\infty)$, $pos(G_\infty)$, $neg(F_\infty)$ and $neg(G_\infty)$ are convex.[1] Theorem 4 says that F_∞ and G_∞ should be distinguishable with any PAC parameters $\epsilon, \delta > 0$, for finite sample-size depending on ϵ, δ. But this will be contradicted by the convergence property of $\{(F_i'', G_i'')\}$.

Define the F-difference between (F_i, G_i) and (F_j, G_j) (denote $d_F((F_i, G_i), (F_j, G_j))$) to be the probability $Pr(F_i(\mathbf{x}) \neq F_j(\mathbf{x}))$ for $\mathbf{x}$ generated by D. We will construct an infinite subsequence $\{(F_i', G_i')\}$ such that for $j > i$,

$$d_F((F_i', G_i'), (F_j', G_j')) < 2^{1-i}.$$

From a result of Pollard [35] (see also Haussler [26]), for any $\zeta > 0$, there is a finite ζ-cover for any collection of sets having finite V-C dimension (which as we have noted in section 1.3 is $d + 1$ in this case). (A ζ-cover of a metric space is a set $\mathcal{K}$ of points such that for all points x in the metric space there is a member of $\mathcal{K}$ within distance ζ of x.)

[1] These regions are not necessarily open or closed halfspaces even if $pos(F_\infty) \cup neg(F_\infty)$ is all of $\mathbf{R}^d$; such a region could for example be $\{(x, y) : x > 0 \vee (x = 0 \wedge y > 0)\}$.

Construct F_i' as follows. Let $F_1' = F_1$. Now construct F_{i+1}' from F_i' maintaining the invariant that there are infinitely many elements of the sequence $\{(F_j, G_j)\}$ which have F-difference $\leq 2^{1-i}$ with (F_i', G_i'). Let $\mathcal{K}_i$ be a finite 2^{-i-1}-cover of the class of linear threshold functions, with respect to input distribution D. Let F_i^- be the (infinitely many) elements of $\{F_j\}$ that differ by $\leq 2^{1-i}$ from F_i'. $\mathcal{K}_i$ must have an element whose 2^{-i-1}-neighborhood contains infinitely many elements of F_i^-. Let F_{i+1}' be one of those elements, and then F_{i+1}' is within 2^{-i} of infinitely many elements of F_i^-. Remove all other elements from the sequence $\{F_j\}$ and continue.

Define the G-difference between (F_i, G_i) and (F_j, G_j) (denote $d_G((F_i, G_i), (F_j, G_j))$) to be the probability $Pr(G_i(\mathbf{x}) \neq G_j(\mathbf{x}))$ for $\mathbf{x}$ generated by D. We may use a similar argument to extract from $\{(F_i', G_i')\}$ an infinite subsequence $\{(F_i'', G_i'')\}$, for which we also have that for $j > i$,

$$d_G((F_i'', G_i''), (F_j'', G_j'')) < 2^{1-i}$$

(as well as $d_F((F_i'', G_i''), (F_j'', G_j'')) < 2^{1-i}$).

We say that a sequence of binary values $(b_1, b_2, b_3 \ldots)$ *converges* when all but a finite number of elements of the sequence are equal. In that case, $\lim_{i\to\infty} b_i$ is well-defined. We prove two properties about the limiting behavior of $\{F_i''\}$ and $\{G_i''\}$.

Claim 1: that with probability 1, a point $\mathbf{x} \in \mathbf{R}^d$ generated by D has the property that the sequences $(F_1''(\mathbf{x}), F_2''(\mathbf{x}), \ldots)$ and $(G_1''(\mathbf{x}), G_2''(\mathbf{x}), \ldots)$ converge.

To prove this claim, consider the probability that for a point $\mathbf{x}$ generated by D, the sequence $(F_N''(\mathbf{x}), F_{N+1}''(\mathbf{x}), \ldots)$ contains more than one binary value, where N is a positive integer. In that case, there must be two consecutive terms that differ. By construction, the probability that $F_i''(\mathbf{x})$ and $F_{i+1}''(\mathbf{x})$ differ is $< 2^{1-i}$, so the probability that any pair of consecutive elements differ is $< \sum_{i=N}^{\infty} 2^{1-i} = 2^{2-N}$. Clearly N can be chosen to make this probability arbitrarily small.

Let $F_\infty(\mathbf{x})$ (resp. $G_\infty(\mathbf{x})$) denote the label assigned to $\mathbf{x}$ by F_i'' (resp. G_i'') for all sufficiently large i. (So F_∞ and G_∞ are well-defined everywhere except on a set of (probability) measure 0.

Claim 2: Let $pos(F_\infty)$ and $neg(F_\infty)$ denote the points which get asymptotic labels 1 and 0 by F_i'', with similar definitions for $pos(G_\infty)$ and $neg(G_\infty)$. We claim that $pos(F_\infty)$, $neg(F_\infty)$, $pos(G_\infty)$, $neg(G_\infty)$ are all convex.

To prove this claim, suppose that $\mathbf{x}$ and $\mathbf{y}$ belong to $pos(F_\infty)$, and let $\mathbf{z}$ be a convex combination of $\mathbf{x}$ and $\mathbf{y}$. Choose N such that the elements of $(F_N''(\mathbf{x}), F_{N+1}''(\mathbf{x}), \ldots)$ and $(F_N''(\mathbf{y}), F_{N+1}''(\mathbf{y}), \ldots)$ are all equal to 1. By convexity of the sets of points which F_i'' labels 1, we deduce that $(F_N''(\mathbf{z}), F_{N+1}''(\mathbf{z}), \ldots)$ are also all equal to 1. Hence $F_\infty(\mathbf{z})$ is well-defined, and so convex combinations of elements of $pos(F_\infty)$ belong to $pos(F_\infty)$. A similar argument hold for the other three sets.

From claim 1 and claim 2, the sets $pos(F_\infty)$, $neg(F_\infty)$, $pos(G_\infty)$ and $neg(G_\infty)$ satisfy the conditions of theorem 4.

Let $M < \infty$ denote a sample size sufficient to distinguish F_∞ from G_∞ with probability $1 - \delta/2$. Choose N sufficiently large such that for random $\mathbf{x}$ generated by D,

$$Pr(F_\infty(\mathbf{x}) = F_i(\mathbf{x})) > 1 - \delta/4M,$$

$$Pr(G_\infty(\mathbf{x}) = G_i(\mathbf{x})) > 1 - \delta/4M,$$

for all $i \geq N$. Then with probability $> 1 - \delta/2$, given M samples, F_i agrees with F_∞ and G_i agrees with G_∞ on those samples, for all $i \geq N$.

Then any method that could distinguish F_∞ from G_∞ with uncertainty $\delta/2$ using M samples can be converted directly to a method to distinguish F_i'' from G_i'' (for all $i \geq N$) with uncertainty at most δ. (In particular replace output of F_∞ with F_i'' and replace output of G_∞ with G_i''.) This contradicts the assumption of monotonic unlimited increase in sample complexity for terms of the sequence $\{(F_i, G_i)\}$. □

2.2 Identifying Polynomial Asymptotic Behavior of Sample Complexity

Regarding point 2 noted at the start of this section, we continue by giving some sufficient conditions on an input distribution to ensure that the asymptotic behavior of sample-size requirements (for pairwise comparisons) is polynomial. Our sufficient conditions for giving polynomial sample complexity use two measures of D defined below, which we denote $V(D)$ and $M(D)$. When these are finite (as they are for many natural continuous distributions) this will imply a lower bound on the difference between means of positive (or negative) examples of pairs of functions that differ by ϵ, and the observed mean can then be used to distinguish the functions, using $poly(\epsilon^{-1})$ examples.

Definition 6 *Given input distribution D, let $V(D)$ denote the largest variance of any of the individual components of vectors generated by D (a quantity which is finite given our assumption of well-defined means and variances for the marginal distributions of D).*

Now let $S(D)$ be the smallest affine linear subspace such that with probability 1, points generated by D lie in that subspace. For a 1-dimensional affine line l in $S(D)$, we can project points generated by D onto l by mapping them to their nearest point on l. Now if points on l are mapped isometrically onto $\mathbf{R}$ by fixing an origin on l and a direction of increase, we have a density p_l over $\mathbf{R}$. Let $M(D)$ denote the maximum (over lines l in $S(D)$ and points in $\mathbf{R}$) of the density p_l. Note that $M(D)$ is infinite if D assigns a non-zero probability to any proper subspace of $S(D)$ (by choosing a line $l \subseteq S(D)$ normal to that subspace).

The measures $M(D)$ and $V(D)$ are motivated by theorem 10 and examples below of distributions for which we give upper bounds on M and V. The following fact is useful later:

Observation 7 *Given any real-valued continuous random variable with an upper bound M on its density, its variance is minimized by making it uniform over an interval of length $1/M$, and the variance is $1/12M^2$. From this we obtain $V(D) \geq 1/12\sqrt{d}M^2$.*

Example 8 *Suppose D_d is uniform over an axis-aligned unit cube in $\mathbf{R}^d$. Then by observation 7, $V(D_d) = 1/12$. To obtain an upper bound on $M(D_d)$, suppose l is a line through the origin, and then points generated by D_d projected onto l can be generated as sums of random variables uniform over $[0, l_i]$ where l_i is the scalar product of a unit vector on l with a unit vector on the i-th axis. The largest of the l_i is $\geq 1/\sqrt{d}$ hence the density is $\leq \sqrt{d}$, so $M(D_d) \leq \sqrt{d}$. More generally, other distributions D for which the measures $M(D)$ and $V(D)$ are well-defined include for example, the uniform distribution over any polytope, including ones of dimension less than d (for which $S(D)$ would be a proper subspace of $\mathbf{R}^d$).*

Example 9 *If D_d is a normal distribution whose covariance matrix is the identity matrix, then $V(D_d) = 1$ and $M(D_d) = (2\pi)^{-1/2}$. More generally, any multivariate normal distribution D also has well-defined $M(D)$ and $V(D)$, even if its covariance matrix does not have full rank. (See for example Von. Mises [34] for standard results about multivariate normal distributions.) For multivariate normal distributions D, $S(D)$ is the space generated by taking the mean of D and adding linear combinations of the eigenvectors of the covariance matrix. $M(D)$ is equal to $(\sigma(2\pi)^{1/2})^{-1}$ where σ^2 is the smallest non-zero eigenvalue of the covariance matrix.*

Theorem 10 *Given any D for which $M(D)$ and $V(D)$ are defined, the sample size required to distinguish any pair (F, G) of linear threshold functions that differ by ϵ (with probability $1 - \delta$) is polynomial in ϵ^{-1} and δ^{-1}, (i.e. the polynomial depends just on D, not on choice of F, G.) In particular, the sample size is $O(\log(\delta^{-1})M(D)V(D)d^{3/2}/\epsilon^2)$.*

Proof: We use the notation introduced in theorem 4:

$$R_{00} = neg(F) \cap neg(G) \qquad R_{01} = neg(F) \cap pos(G)$$
$$R_{10} = pos(F) \cap neg(G) \qquad R_{11} = pos(F) \cap pos(G)$$

The region of disagreement is $R_{01} \cup R_{10}$, and we are assuming that

$$D(R_{01}) + D(R_{10}) = \epsilon.$$

We may assume that in addition we have

$$D(R_{01}) \geq \epsilon/4, \quad D(R_{10}) \geq \epsilon/4$$

since otherwise for F and G there is a difference of at least $\epsilon/2$ that a random example is positive, and F and G could be distinguished with $O(\epsilon^{-1}\log(\delta^{-1})$ examples using that property.

As before let $\mu(R_{01})$ and $\mu(R_{10})$ denote the expectations of points lying in these regions. The marginal variances of points generated by D are upper-bounded by $V(D)$, so given a sufficient distance between the means of R_{01} and R_{10}, we should be able to use the observed means of the positive (or negative) examples to distinguish F from G with high confidence. We claim that there is a lower bound on the Euclidean distance $|\mu(R_{01}) - \mu(R_{10})|$ which depends on $M(D)$ and $V(D)$, but not F or G, and is polynomial in ϵ^{-1}.

Suppose for a contradiction that

$$|\mu(R_{01}) - \mu(R_{10})| < \frac{\epsilon}{16M(D)}.$$

Let l be a 1-dimensional line that is normal to the hyperplane spanned by $\mu(R_{00})$ and the intersection of the hyperplanes defining F and G.

For $R \subseteq \mathbf{R}^d$ let $l(R)$ denote the set of points on l that are closest to some point in R (the projection of R onto l). Then $l(R_{01}) \cap l(R_{10}) = \emptyset$, but

$$|l(\{\mu(R_{01})\}) - l(\{\mu(R_{10})\})| < \frac{\epsilon}{16M(D)}.$$

By Markov's inequality, for random $\mathbf{x} \in R_{01}$ ($\mathbf{x}$ generated by D restricted to R_{01}),

$$Pr\Big(|l(\{\mathbf{x}\}) - l(\{\mu(R_{01})\})| < \frac{\epsilon}{16M(D)}\Big) > 1/2$$

(and similarly for points in R_{10}.) Hence the probability of points in the range $[l(\{\mu(R_{01})\}) - \frac{\epsilon}{16M(D)}, l(\{\mu(R_{01})\}) + \frac{\epsilon}{16M(D)}]$ is greater than $\frac{1}{2}(\frac{\epsilon}{4})$ *i.e.* the density is greater than $\frac{1}{2}(\frac{\epsilon}{4})/(\epsilon/8M(D)) = M(D)$, a contradiction.

So we conclude that the distance between the points $l(\mu(R_{01}))$ and $l(\mu(R_{10}))$ is at least $\epsilon/(16M(D))$. By construction of l, we also have that $l(\mu(R_{00}))$ lies between $l(\mu(R_{01}))$ and $l(\mu(R_{10}))$. We use this observation to deduce a lower bound on the distance between $l(\mu(neg(F)))$ and $l(\mu(neg(G)))$. $l(\mu(neg(F)))$ is a weighted average of $l(\mu(R_{00}))$ and $l(\mu(R_{01}))$, and the weight for $l(\mu(R_{01}))$ is at least $\epsilon/4$ (and similarly for $l(\mu(neg(G)))$ with respect to R_{10} instead of R_{01}). Hence the distance between $l(\mu(neg(F)))$ and $l(\mu(neg(G)))$ is at least $(\frac{\epsilon}{4})\epsilon/(16M(D)) = \epsilon^2/(64M(D))$.

It follows that the Euclidean distance between $\mu(neg(F))$ and $\mu(neg(G))$ is also at least $\epsilon^2/(64M(D))$. Hence in some component, the distance between these means is at least $\epsilon^2/(64M(D)\sqrt{d})$. The marginal variances are all upper-bounded by $V(D)$, so the number of observations of that component's value needed to identify which of the two alternative means is correct with probability $1 - \delta$, is $O(\log(\delta^{-1})V(D)M(D)\sqrt{d}/\epsilon^2)$. Given that each component is equally likely to be observed, the overall sample complexity becomes $O(\log(\delta^{-1})V(D)M(D)d^{3/2}/\epsilon^2)$. □

$M(D)$ and $V(D)$ are crude measures in that for distributions D for which they are large, the actual sample size needed may not be correspondingly large. We consider the question of when a similar result should exist for probability distributions D which do not satisfy the condition of theorem 10. For example, finite unions of point probability masses are of interest, but automatically do not have finite $M(D)$.

Corollary 11 *Suppose D is*

1. *a finite union of point probability masses, or, more generally,*
2. *a mixture of a finite union of point probability masses and a distribution D' for which $M(D')$ and $V(D')$ are finite*

then the sample size needed to distinguish F and G (defined in the same way as in theorem 10) is polynomial in the PAC parameters, and independent of F, G.

Proof: It is straightforward to prove part 1 of this result, it is in fact a slight generalization of the argument of Chow [17]. Let $\alpha > 0$ be the smallest weight assigned to any of the point probability masses. Clearly if $F \neq G$ then they must have probability at least α of disagreement.

Since there are only finitely many points in the domain of D, there are only finitely many pairs of distinct linear threshold functions. Hence there is a non-zero lower bound on the difference between the means of positive examples of F and of G. By the proof of theorem 10 this provides a sample complexity that is polynomial in ϵ^{-1} and δ^{-1}, and independent of any other features of F and G.

For the extension to part 2 of this result, again let α be the smallest weight of any of the point probability masses, and then for F and G which differ by $\epsilon < \alpha$, their behavior on points generated by D' will distinguish them (since they cannot disagree on any of the point probability masses). Since $M(D')$ and $V(D')$ are finite, the sample complexity is polynomial, by theorem 10.

For any $\epsilon > \alpha$, put $\delta = 1/4$ and by corollary 5 there exists a finite positive sample size $m(\epsilon, D)$ sufficient to distinguish any pair F, G which differ by ϵ. Let $M = \max_{\epsilon \in [\alpha,1]} m(\epsilon, D)$, which must be finite, since otherwise we would have a positive ϵ for which the sample complexity is infinite. Use a sample size of M for $\epsilon > \alpha$. For smaller values of δ we can obtain sample complexity logarithmic in δ^{-1} by taking the majority vote of a logarithmic (in δ^{-1}) number of hypotheses which have confidence parameter $1/4$. □

We suspect the set of "good" distributions should be generalizable further; see section 5.

2.3 Input distributions which lead to super-polynomial Sample Complexity

Informed by the sufficient conditions identified for polynomial behavior, we next define a distribution which does not give rise to polynomial behavior. That is, for any function f, we can construct rather artificial input distributions that cause at least $f(\epsilon^{-1})$ 1-RFA examples to be needed to distinguish certain pairs of linear threshold functions that differ by ϵ, for all $\epsilon > 0$.

Theorem 12 *Let f be some arbitrary increasing function. There exists a bounded input distribution $D(f)$ on $\mathbf{R}^3$ such that for all ϵ there exists a pair of linear threshold functions*

which differ by ϵ and require at least $f(\epsilon^{-1})$ samples to be distinguishable with confidence $1-\delta$, for $\delta < 1/2$.

Proof: The domain of D is restricted to a sequence of pairs of line segments (l_i, l'_i) defined as follows. All the line segments are parallel to the line given by $x = y = z$, are of unit length, and have endpoints in the planes given by $x+y+z = 0$ and $x+y+z = \sqrt{3}$. We define their exact locations with reference to a set of planes defined as follows.

Define P to be a plane containing the line $x = y = z$, and let F be a linear threshold function with threshold P. Let P_i, $i \in \mathbf{N}$, denote a sequence of planes containing $x = y = z$, such that their angles with P converge to 0 monotonically. (see figure 3. The point of intersection of the lines in figure 3 represents the line $x = y = z$.) The sequence P_i defines a sequence of linear threshold functions G_i such that the symmetric difference of $pos(G_i)$ and $pos(F)$ strictly contains the symmetric difference of $pos(G_j)$ and $pos(F)$, for all $j > i$.

The locations of line segments l_i, l'_i are specified as follows.

l_0 lies in $neg(F) \cap neg(G_0)$.
For $i \geq 1$, l_i lies in $(neg(F) \cap neg(G_i)) \setminus neg(G_{i-1})$.
l'_0 lies in $pos(F) \cap pos(G_0)$.
For $i \geq 1$, l_i lies in $(pos(F) \cap pos(G_i)) \setminus pos(G_{i-1})$.

Finally, the distances of l_i and l'_i from the line $x = y = z$ are constrained to be $1/2f(2^i)$, where f is as defined in the statement of this theorem.

We complete our definition of D by assigning probability $2^{-(1+i)}$ to $l_i \cup l'_i$, and that probability is uniformly distributed over those two line segments.

Given this definition of D, we now claim that for target error ϵ, we need to observe $f(\epsilon^{-1})$ random 1-RFA examples from D in order to distinguish F from an alternative hypothesis G_i chosen such that i is as large as possible subject to the constraint that F disagrees with G_i with probability at least ϵ.

The region of disagreement of F with G_i is the union $\cup_{j=i+1}^{\infty}(l_j \cup l'_j)$, so examples from this set of line segments need to be used in order to distinguish F from G_i. But we now observe that (by analogy with the construction of example 2) with high probability, any example generated from this region has the same conditional likelihood for F as for G_i. In particular, for any point on l_j $(j > i)$ that is more distant than $1/f(\epsilon^{-1})$ from an endpoint of l_j, for any value observed for one of its 3 coordinates, there exists a corresponding point on l'_j which has equal likelihood of generating the same single-coordinate observation. However points on l_j and l'_j should receive opposite labels from F and from G_i, for $j > i$. So with probability at least $1 - 1/f(\epsilon^{-1})$ D fails to generate a point that distinguishes F from G_i. □

The "bad" input distribution defined above has marginal distributions on the input components x, y and z which have well-defined means and variances (this is obvious from the fact that the distribution is defined on a bounded region of the domain $\mathbf{R}^3$). If we dispense with the requirement of well-defined means and variances, then we can define similar "bad" distributions in two dimensions, as follows.

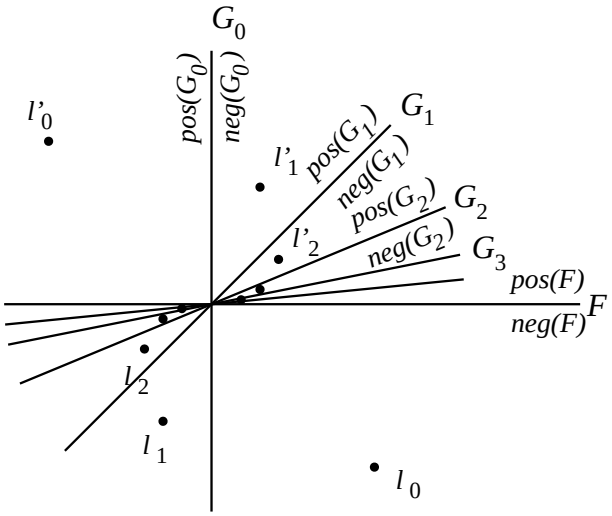

figure 3

construction of theorem 12 shown in cross − section
using plane given by $x + y + z = 0$

The domain of D is restricted to the two lines $y = x$ and $y = x + 1$, for positive values of x and y. As in the statement of theorem 12, let f be an arbitrary increasing function, and we define a bad distribution D associated with f as follows. For $i \in \mathbf{N}$, let D be locally uniform over pairs of line segments whose x-coordinates lie in the range

$$R_i = [\sum_{r=1}^{i} f(2^i), \sum_{r=1}^{i+1} f(2^{i+1})]$$

We let the probability that a random example lies in R_i be given by $D(R_i) = 2^{-i-1}$.

Now we can define two linear threshold functions F and G (see figure 4) which disagree on the intervals whose x-coordinates lie in R_i and agree elsewhere. We can now argue in a similar way to before that single-coordinate observations from these regions (the ones which should allow us to distinguish F from G) have (with probability at least $1 - 1/f(\epsilon)$) equal likelihood for both functions, where i is chosen to minimize 2^{-i} subject to $\epsilon \leq 2^{-i}$.

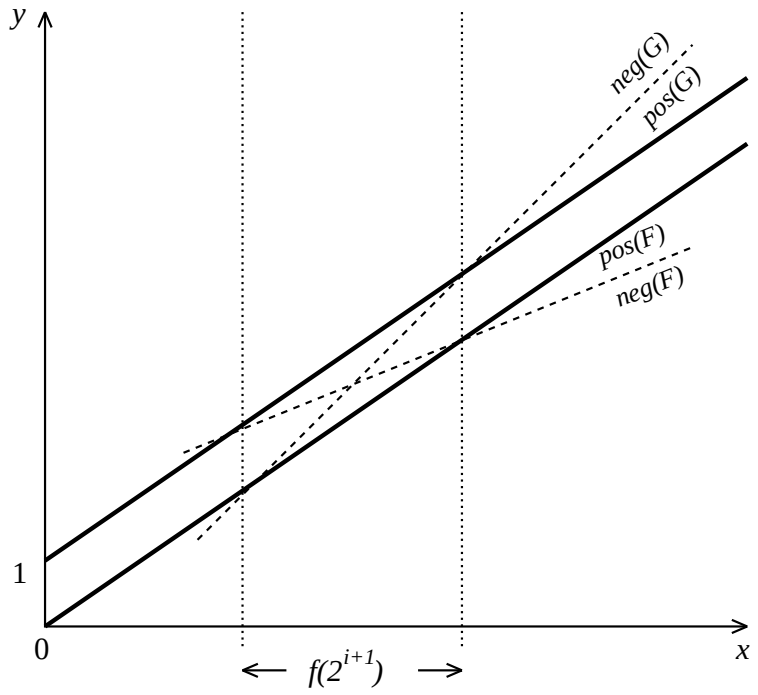

figure 4

The domain is restricted to the two heavy lines. F and G disagree on points occurring between the two vertical dotted lines. This region of disagreement has probability 2^{-i}.

3 A PAC Algorithm

In this section we give a PAC learning algorithm whose runtime is polynomial in ϵ^{-1} and δ^{-1} provided D has finite measures $M(D)$ and $V(D)$, or satisfies corollary 11. Moreover if we have a class of distributions D_d over $\mathbf{R}^d$, $d = 1, 2, 3, \ldots$, for which $M(D_d)$ and $V(D_d)$ are polynomial in d (for example the sequences of distributions in examples 8 and 9) then the algorithm has sample complexity polynomial in ϵ^{-1}, δ^{-1} and d, but the runtime is exponential in d. We start by describing the algorithm, then give results to justify the steps. The algorithm is initially presented in the standard PAC setting. In section 3.3 we show how to express it as a "statistical query" algorithm, as introduced by Kearns [28], who showed that such algorithms are noise-tolerant. First we need the following definition.

Definition 13 *The* quadratic loss *[30] of an example (x, l) (with respect to a classifier F) where x is the input and l is a binary valued label, is the quantity $(l - Pr(label = 1 \mid x; F))^2$, i.e. the square of the difference between l and the probability that F would assign label* 1 *to input x.*

In our case x consists of a real value that has been assigned to a single (known, randomly-chosen) component of a vector $\mathbf{x}$ in the domain $\mathbf{R}^d$, where $\mathbf{x}$ was generated by D.

3.1 The Algorithm

1. Put $\zeta = \epsilon^8/6 \cdot 2^{44} d^2 M(D)^4 V(D)^2$, where ϵ is the target error. Generate a set S of $\Theta(d \log(\delta^{-1})/\zeta^3)$ (unlabeled) points in $\mathbf{R}^d$ from the input distribution D. Thus

$$|S| = \Theta(d^7 \log(\delta^{-1}) M(D)^{12} V(D)^6 / \epsilon^{24}).$$

2. Generate a set $\mathcal{H}$ of candidate hypotheses using the standard method of [11] (see below), such that for each binary labeling of S consistent with a linear threshold function, $\mathcal{H}$ contains exactly one linear threshold function that induces that labeling.

3. Generate a set of labeled 1-RFA data and for each member $H \in \mathcal{H}$, use that data set to estimate the expected quadratic loss of 1-RFA data with respect to H (the average over all examples of their quadratic losses). We show that a sufficiently large sample size for this step has an order of growth dominated by the expression for $|S|$, above.

4. Output the member of $\mathcal{H}$ with the smallest quadratic loss as observed on the 1-RFA data.

The method of [11] works as follows. Let $S = \{x_1, \ldots, x_m\}$. The set of all sequences of labels consistent with the first i elements of S is constructed inductively from the set consistent with the first $i-1$ elements as follows. For each sequence of labels consistent with $\{x_1, \ldots, x_{i-1}\}$, check whether each of the two possible extensions of that label sequence to a sequence of i labels, is consistent with $\{x_1, \ldots, x_i\}$. If so, add that label sequence to the collection that is consistent with the first i elements. This method just requires that it be possible to efficiently test whether a function in the class of interest is consistent with a particular set of labeled data, which is of course possible for linear threshold functions in fixed dimension. Finally, for each label sequence for the entire set S, return a consistent function (in our case, a linear threshold function).

Regarding step 3, in the standard PAC framework we can use the empirical estimate for the quadratic loss, and in section 3.2 we prove that the sample size used above is sufficient. In section 3.3 we show how step 3 can be done using statistical queries, which shows that the algorithm can be made robust to a uniform misclassification noise process.

3.2 Justification of the Algorithm

Using results of Bartlett et al. [3] we can say that $\mathcal{H}$ is an *empirical ζ-cover* of the set of linear threshold functions (where ζ is given by the expression in step 1 of the algorithm). An empirical ζ-cover of a class $\mathcal{C}$ of functions is a subset of $\mathcal{C}$ constructed as follows from a sufficiently large sample S of unlabeled points: for each binary labeling of S consistent

with some element of $\mathcal{C}$, include an arbitrary member of $\mathcal{C}$ that induces that labeling. It is shown in [3], that with high probability the resulting set $\mathcal{H}$ contains, for any $F \in \mathcal{C}$, a member that differs from F by at most ζ. In particular, it is shown that if a sample of size m is randomly generated, then the probability that two functions exist whose observed disagreement on the sample differs from their true disagreement by more than $\zeta/2$, is upper-bounded by

$$16\Big(\frac{1024}{\zeta}\Big)^{12d\log_2(2056em/(d\zeta))}e^{-\zeta^2 m/(512)}.$$

Below we verify that this can be upper-bounded by δ if $m = \Theta(d\log(\delta^{-1})/\zeta^3)$. (Note that in the bounds of [3], d is the value of the fat-shattering function with parameter ζ, which for binary classifiers is equal to the V-C dimension, for any ζ.)

Inserting $m = \Theta(d\log(\delta^{-1})/\zeta^3)$ in the above expression, we find that its order of growth is

$$\Big(\frac{1}{\zeta}\Big)^{d\log(\log(\delta^{-1})/\zeta^4)}e^{-d\log(\delta^{-1})/\zeta}$$

$$< \Big(\frac{1}{\zeta}\Big)^{d\log(\delta^{-1}/\zeta^4)}(\delta)^{d/\zeta} = \Big(\frac{1}{\zeta}\Big)^{d\log(\delta^{-1}/\zeta^4)}(\delta)^{(d/\zeta)-1}\delta$$

$$= \Big(\frac{1}{\zeta}\Big)^{d\log(\delta^{-1}/\zeta^4)}\Big(\frac{1}{\zeta}\Big)^{(-\log_\zeta(\delta))((d/\zeta)-1)}\delta = \Big(\frac{1}{\zeta}\Big)^{[d\log(\delta^{-1}/\zeta^4)+\log_\zeta(\delta)-(d/\zeta)\log_\zeta(\delta)]}\delta$$

which for ζ, δ below some constant (depending on the base of the logarithms in the above expression) is equal to δ multiplied by a value in the range $[0, 1]$. (Observe that the value is $1/\zeta$ raised to the power of a negative quantity.)

The next part of the algorithm finds the hypothesis with the smallest quadratic loss. Since our set of candidate hypotheses is of polynomial size, we could just find an optimal one using pairwise comparisons. Our reasons for preferring to use quadratic loss are firstly that we have the problem that the set $\mathcal{H}$ of candidate functions does not generally contain the target function; so far our results for pairwise comparison have assumed that one of the functions being compared is the target. The second reason is that minimizing the quadratic loss seems potentially more amenable to heuristics for optimization over an exponentially large set of candidate hypotheses (eg. when d is not constant).

We can use results of [30] to claim that minimizing quadratic loss is a good strategy. For our purposes quadratic loss is a good loss function for the following two reasons.

1. Like the negative log likelihood loss function, the expected quadratic loss of a hypothesis is minimized when hypothesis conditional probabilities equal the true conditional probabilities.

2. Unlike the negative log likelihood, quadratic loss is bounded (takes values in $[0, 1]$), so automatically we have a guarantee that (with high probability) observed expected quadratic loss converges quickly to true expected quadratic loss.

(The disadvantage of quadratic loss by comparison with negative log likelihood is that it may only be used for 2-class classification, which is what we have here.)

Notation: For a classifier F let $QL(F)$ denote its expected quadratic loss (on random examples assumed to be labeled by some target function) and let $\hat{QL}(F)$ denote observed expected quadratic loss for some sample of labeled points. We have noted that $\hat{QL}(F)$ converges reasonably quickly to $QL(F)$, since quadratic loss is bounded (lies in $[0,1]$). We also need to be able to claim that if F is any target function we have:

1. If F and G differ by ϵ (so G has error ϵ), then $QL(G) - QL(F)$ is upper bounded by some polynomial in ϵ
2. If F and G differ by ϵ, then $QL(G) - QL(F)$ is lower bounded by some other polynomial in ϵ

These two properties will validate the approach of finding minimal quadratic loss over members of an ζ-cover, for ζ^{-1} polynomial in ϵ^{-1}. Regarding the first, we show below that $QL(G) - QL(F) \leq 3\epsilon$. Theorem 19 will prove the second. Finally, theorem 21 uses these properties and also shows that although we do not have the exact values of quadratic loss for members of the ϵ-cover, we can still estimate them well enough for our purposes in polynomial time.

Proposition 14 *If F is the target function and G has error ϵ, then $QL(G) - QL(F) \leq 3\epsilon$.*

Proof: Let X' be the part of the domain X where F and G agree. The points in $X \setminus X'$ contribute at most ϵ to the difference $QL(G) - QL(F)$, since quadratic loss lies in the range $[0,1]$. We upper bound by 2ϵ the contribution of points in X'.

Let $x_1, \ldots, x_d$ be the components of input vectors, each one of which is observed with probability $1/d$. Let $D(X')|_{x_i}$ denote the distribution of x_i-coordinates of elements of X'. For points in X' with binary label l, the difference $QL(G) - QL(F)$ between expected quadratic losses is:

$$\sum_{i=1}^{d} \frac{1}{d} \int_{r \in \mathbf{R}} \Big[\Big(Pr(label = 1 \mid x_i = r; G) - l \Big)^2 - \Big(Pr(label = 1 \mid x_i = r; F) - l \Big)^2 \Big] D(X')|_{x_i}(r) dr$$

Putting $l = 0$ or alternatively $l = 1$ into the above expression, and noting that the difference between the squares of two numbers in the range $[0,1]$ is upper bounded by twice their difference, the above is upper bounded by:

$$\sum_{i=1}^{d} \frac{1}{d} \int_{r \in \mathbf{R}} 2 \Big| Pr(label = 1 \mid x_i = r; G) - Pr(label = 1 \mid x_i = r; F) \Big| D(X')|_{x_i}(r) dr$$

$$= \sum_{i=1}^{d} \frac{2}{d} \int_{r \in \mathbf{R}} \epsilon D(X')|_{x_i}(r) dr = 2\epsilon.$$

□

We now move on to proving the second claim, that there is a polynomial lower bound on the difference in quadratic loss.

Definition 15 *The variation distance between two probability distributions D, D' over* $\mathbf{R}$ *is defined to be*

$$var(D, D') = \int_{r \in \mathbf{R}} |D(r) - D'(r)| dr$$

Our strategy to prove theorem 19 is to relate error of a hypothesis G (for target F) to the variation distance between the marginal distributions on some input component x of its positive (respectively, negative) examples, and the marginal distributions on x of the positive (respectively, negative) examples of F (lemma 16). Then the variation distance is related to expected quadratic loss using lemma 17 in conjunction with lemma 18. We assume throughout that continuous densities $D(r)$ and $D'(r)$ are Lebesgue integrable, so that it follows that $|D(r) - D'(r)|$ and $\max\{0, D'(r) - D(r)\}$ are also Lebesgue integrable (and integrate to $var(D, D')$ and $\frac{1}{2}var(D, D')$ respectively over $\mathbf{R}$).

Lemma 16 *Let D and D' be two probability distributions over* $\mathbf{R}$*, such that the difference between their means is μ and their variances are both upper-bounded by σ^2. Then their variation distance $var(D, D')$ is at least* $\min\{1, (\mu/\sigma)^2/8\}$.

Proof: We may assume that the mean of D is 0 and the mean of D' is $\mu \geq 0$. We obtain an upper bound on μ in terms of $var(D, D')$ and σ^2, and convert that result into a lower bound on $var(D, D')$ in terms of μ and σ^2.

Define distribution D'' as follows:

$$D''(r) = \frac{2}{var(D, D')} \max\{0, D'(r) - D(r)\}.$$

The coefficient $\frac{2}{var(D,D')}$ normalizes D'' — we are assuming of course that $var(D, D') > 0$. If $var(D, D') = 0$ then $\mu = 0$ and the result holds. The following procedure samples from D':

1. sample $r \in \mathbf{R}$ from D. If $D(r) \leq D'(r)$, accept r. Otherwise proceed to step 2.
2. (We have $D(r) > D'(r)$.) Accept r with probability $D'(r)/D(r)$, else reject r.
3. If r was rejected above, sample from D''.

Observe that the probability that r is rejected in step 2 is $\frac{1}{2}var(D, D')$. Let s be the expected value of rejected points. The upper-bound on variance of D gives an upper bound on the (absolute value of the) expected value of rejected points as follows:

$$\sigma^2 \geq s^2 . \frac{1}{2} var(D, D')$$

Rearranging to get an upper bound on $|s|$:

$$|s| \leq \sigma \sqrt{2/var(D, D')}.$$

Now μ is equal to the rejection probability $\frac{1}{2}var(D, D')$, multiplied by the expected value of points sampled from D'' minus the expected value of rejected points, *i.e.*

$$\mu = \frac{1}{2}var(D, D')\Big(E(D'') - s\Big).$$

Again using the upper bound on variance, this time variance of D':

$$E(D'') - \mu \leq \sigma\sqrt{2/var(D, D')}.$$

Combining the two expressions above we have

$$\mu \leq \frac{1}{2}var(D, D')\Big(\sigma\sqrt{2/var(D, D')} + \mu - s\Big)$$

Using our upper bound for $|s|$ (in particular $-s \leq \sigma\sqrt{2/var(D, D')}$) and rearranging,

$$\mu(2 - var(D, D')) \leq var(D, D')2\sigma\sqrt{2/var(D, D')}.$$

Rearranging the above,

$$\mu \leq \frac{2^{3/2}\sigma[var(D, D')]^{1/2}}{2 - var(D, D')}$$

Provided that $var(D, D') \leq 1$ we have

$$\mu \leq 2^{3/2}\sigma[var(D, D')]^{1/2}$$

Hence

$$var(D, D') \geq (\mu/\sigma)^2/8 \quad \text{or} \quad var(D, D') > 1$$

□

Lemma 17 *Let F be the target linear threshold function and G some other linear threshold function. Let $D(pos(F))|_x$, $D(neg(F))|_x$, $D(pos(G))|_x$, $D(neg(G))|_x$, be the distributions of the x component of positive and negative examples of F and G. For $R \subseteq \mathbf{R}^d$, let $D(R)$ denote the probability that a random input vector lies in R. Suppose that we have*

$$\begin{aligned} var(D(pos(F))|_x, D(pos(G))|_x) &> \epsilon \\ D(pos(F)) &> 1/4. \\ D(pos(G)) &> 1/4. \end{aligned}$$

Then for 1-RFA data for which x is the observed component, we have a lower bound of $\epsilon/32$ on the expected difference between the conditional probabilities of output label 1 for F and G, for random values of x. We can make the same deduction from the above assumptions applied to $neg(F)$ and $neg(G)$ instead of $pos(F)$ and $pos(G)$.

Proof: We prove this by contradiction. Suppose for a contradiction that for $r \in \mathbf{R}$ distributed according to $D|_x$, the marginal distribution of D on x, that

$$E\Big(\Big|Pr(label = 1 \mid x = r; F) - Pr(label = 1 \mid x = r; G)\Big|\Big) < \epsilon/32.$$

Then we have

$$\Big|D(pos(F)) - D(pos(G))\Big| < \epsilon/32.$$

We have assumed for contradiction that

$$\int_{r\in\mathbf{R}} \Big|Pr(label = 1 \mid x = r; F) - Pr(label = 1 \mid x = r; G)\Big| D|_x(r)dr < \epsilon/32$$

(where $D|_x$ is the marginal distribution of D on x.)

Observe that $D(pos(F))|_x(r) = D|_x(r)\frac{Pr(label=1 \mid x=r;F)}{D(pos(F))}$ and similarly for G. Hence the variation distance $var(D(pos(F))|_x, D(pos(G))|_x)$ is equal to

$$\int_{r\in\mathbf{R}} \Big|\frac{Pr(label = 1 \mid x = r; F)}{D(pos(F))} - \frac{Pr(label = 1 \mid x = r; G)}{D(pos(G))}\Big| D|_x(r)dr$$

$$\leq 4\int_{r\in\mathbf{R}} \Big|Pr(label = 1 \mid x = r; F) - \frac{D(pos(F))}{D(pos(G))} Pr(label = 1 \mid x = r; G)\Big| D|_x(r)dr$$

$$\leq 4\int_{r\in\mathbf{R}} \Big|Pr(label = 1 \mid x = r; F) - Pr(label = 1 \mid x = r; G)\Big| D|_x(r)dr$$

$$+ 4\int_{r\in\mathbf{R}} \Big|\Big(1 - \frac{D(pos(F))}{D(pos(G))}\Big)\Big| Pr(label = 1 \mid x = r; G) D|_x(r)dr$$

$$\leq 4\Big(\frac{\epsilon}{32}\Big) + 4\Big|\Big(1 - \frac{D(pos(F))}{D(pos(G))}\Big)\Big| \leq \frac{\epsilon}{8} + 4\Big(\frac{\epsilon}{8}\Big) < \epsilon,$$

which contradicts one of the assumptions made in the lemma.

We have established the lower bound of $\epsilon/32$. □

Lemma 18 *Let F be the target function and G some other function and suppose that ϵ is the expected difference between the conditional probabilities of output 1 for F and G, over random inputs from input distribution D. Then*

$$QL(G) - QL(F) \geq \epsilon^2.$$

Proof: Let x be an input component, and suppose that for some 1-RFA input $x = r$, we have

$$\begin{aligned} Pr(label = 1 \mid x = r; F) &= p, \\ Pr(label = 1 \mid x = r; G) &= p + \xi. \end{aligned}$$

Then the expected quadratic loss of F for input $x = r$ is

$$QL(F \mid x = r) = p(1-p)^2 + (1-p)p^2$$

For G we have

$$\begin{aligned} QL(G \mid x = r) &= p(1-p-\xi)^2 + (1-p)(p+\xi)^2 \\ &= p(1-p)^2 + (1-p)p^2 + \xi^2 \\ &= QL(F \mid x = r) + \xi^2 \end{aligned}$$

By convexity, the expected quadratic loss of G averaged over random input values is minimized by assuming that for all $r \in \mathbf{R}$, the difference in conditional probabilities is uniform, so that for any input $x = r$,

$$\Big|Pr(label = 1 \mid x = r; G) - Pr(label = 1 \mid x = r; F)\Big| = \epsilon.$$

So for inputs consisting of observations of x, the difference between expected quadratic losses of G and F is at least ϵ^2. □

We now use all these lemmas in the following

Theorem 19 *For the class of linear threshold functions over $\mathbf{R}^d$, suppose that the input distribution D has finite values $M(D)$ and $V(D)$ as defined in definition 6, and that the target function has quadratic loss Q^*. Then any function with error $\epsilon < 1/4$ has quadratic loss at least $Q^* + p(\epsilon)$ for polynomial p where*

$$p(\epsilon) = \frac{\epsilon^8}{2^{44} d^2 . M(D)^4 V(D)^2}.$$

Proof: Let F be the target and let G be another function with error ϵ. We consider two cases:

1. for random $\mathbf{x} \in \mathbf{R}^d$, $|Pr(F(\mathbf{x}) = 1) - Pr(G(\mathbf{x}) = 1)| > \epsilon/2$
2. for random $\mathbf{x} \in \mathbf{R}^d$, $|Pr(F(\mathbf{x}) = 1) - Pr(G(\mathbf{x}) = 1)| \leq \epsilon/2$

Case 1: for any input component x,

$$\int_{r \in \mathbf{R}} \Big|Pr(label = 1 \mid x = r; F) - Pr(label = 1 \mid x = r; G)\Big| . D|_x(r) dr > \epsilon/2.$$

Hence by lemma 18,

$$QL(G) - QL(F) > \epsilon^2/4.$$

Case 2: we use the notation introduced in theorem 4:

$$\begin{aligned} R_{00} &= neg(F) \cap neg(G) \qquad R_{01} = neg(F) \cap pos(G) \\ R_{10} &= pos(F) \cap neg(G) \qquad R_{11} = pos(F) \cap pos(G) \end{aligned}$$

The region of disagreement is $R_{01} \cup R_{10}$, and by the assumption of the theorem,

$$D(R_{01}) + D(R_{10}) = \epsilon.$$

In addition, from the assumption of case 2:

$$D(R_{01}) \geq \epsilon/4, \quad D(R_{10}) \geq \epsilon/4.$$

Assume that $D(pos(F)) > 1/4$ and $D(pos(G)) > 1/4$. (If not we would have $D(neg(F)) > 1/4$ and $D(neg(G)) > 1/4$, and that case would be handled similarly to what follows.) We continue by lower-bounding $|\mu(R_{01}) - \mu(R_{10})|$, from which we get a lower bound on the difference between the means of 1-labeled examples of F and G, and we also have an upper bound on their variances. From these we get a lower bound for $var(D(pos(F))|_x, D(pos(G))|_x)$ for some component x, then use lemmas 17 and 18 to get the lower bound on quadratic loss.

From the proof of theorem 10 we have

$$|\mu(R_{01}) - \mu(R_{10})| \geq \frac{\epsilon}{16M(D)}$$

from which we showed how to deduce

$$|\mu(pos(F)) - \mu(pos(G))| \geq \frac{\epsilon^2}{64M(D)}.$$

Let $\mu(R)|_x$ and $\sigma^2(R)|_x$ denote the expectation and variance of x-coordinates of points generated by D that lie in $R \subseteq \mathbf{R}^d$. For some component x:

$$\Big|\mu(pos(F))|_x - \mu(pos(G))|_x\Big| \geq \frac{\epsilon^2}{64\sqrt{d}M(D)}.$$

We also have from our assumption $D(pos(F)) > 1/4$ and $D(pos(G)) > 1/4$ and the assumed upper bound on the marginal variances of D:

$$\sigma^2(pos(F))|_x \leq 4V(D), \qquad \sigma^2(pos(G))|_x \leq 4V(D).$$

Hence using lemma 16 we have that the variation distance between the x-value of points lying in $pos(F)$ and points lying in $pos(G)$ is at least

$$\min\Big\{1, \frac{\epsilon^4/2^{12}d.M(D)^2}{8(4V(D))}\Big\}$$

$$= \min\Big\{1, \frac{\epsilon^4}{2^{17}d.M(D)^2V(D)}\Big\} = \frac{\epsilon^4}{2^{17}d.M(D)^2V(D)}$$

using observation 7 and the fact that $\epsilon \leq 1$.

Hence the expected difference between conditional probabilities of output 1 for F and G is by lemma 17, at least

$$\frac{\epsilon^4}{2^{22}d.M(D)^2V(D)}.$$

Finally, we use lemma 18 to obtain

$$QL(G) - QL(F) \geq \frac{\epsilon^8}{2^{44}d^2.M(D)^4V(D)^2}.$$

The lower bound of case 2 can be seen to be strictly weaker than the lower bound for case 1, so the combination is just the lower bound for case 2. □

We omit the proof of the following result.

Theorem 20 *For the class of linear threshold functions over $\mathbf{R}^d$, suppose that the input distribution D satisfies the criteria of corollary 11, and that the target function has quadratic loss Q^*. Then any function with error ϵ has quadratic loss at least $Q^* + p(\epsilon)$ for some positive increasing polynomial p.*

This extension to the weaker constraints of corollary 11 just involves bounding the means of the regions of disagreement away from each other (as done in the proofs of theorem 10 and corollary 11) and then proceeding as in the above proof.

We have now shown how the expected quadratic loss of a hypothesis is polynomially related to its disagreement with the target function. The following result uses this relationship to justify the strategy of finding a hypothesis of minimal quadratic loss (over a ζ-cover $\mathcal{K}$ that may not necessarily contain the target function), as well as showing that the *observed* quadratic losses of elements of $\mathcal{K}$ are sufficiently good estimates of the true quadratic losses.

Theorem 21 *Let $\mathcal{C}$ be a set of binary classifiers with V-C dimension d, and let QL be the quadratic loss function as defined earlier. Suppose that there are positive increasing polynomials p, p' such that if any $F \in \mathcal{C}$ has error α, we have*

$$Q^* + p(\alpha) \leq QL(F) \leq Q^* + p'(\alpha)$$

(where Q^* is the quadratic loss of the target function.) Then the strategy of minimizing the observed quadratic loss over an empirical ζ-cover achieves PAC-ness, for $\zeta = p'^{-1}(\frac{1}{2}p(\epsilon))$ and polynomial sample size.

Comment: The result would hold for any loss function that had the associated polynomials p and p'. We have shown in theorem 19 that a suitable p exists for the quadratic loss function, and from proposition 14 for quadratic loss we can put $p'(\alpha) = 3\alpha$.

Proof: Let $\zeta = p'^{-1}(\frac{1}{2}p(\epsilon))$, so ζ^{-1} is polynomial in ϵ^{-1}. Let $\mathcal{K}$ be the ζ-cover. We have $|\mathcal{K}| = O((d\log(\delta^{-1})/\zeta^3)^d)$, and we used $O(d\log(\delta^{-1})/\zeta^3)$ unlabeled examples to generate it.

Let $F \in \mathcal{K}$ have error $\leq \zeta$. Then

$$QL(F) \leq Q^* + p'(\zeta) = Q^* + \frac{1}{2}p(\epsilon)$$

Let $G \in \mathcal{K}$ have error $> \epsilon$. Then

$$QL(G) \geq Q^* + p(\epsilon)$$

Now choose a sufficiently large sample such that with probability $1 - \delta$, the observed expected quadratic loss of each element of $\mathcal{K}$ is within $p(\epsilon)/4$ of its true expected quadratic loss. (This ensures that the choice of smallest observed quadratic loss is not a hypothesis with error $> \epsilon$.) We will identify a sample size that ensures this will hold for all members of $\mathcal{K}$.

Let $\gamma = \delta/|\mathcal{K}|$. We want a sample size large enough such that with probability $1 - \gamma$ any given element of $\mathcal{K}$ has observed expected quadratic loss within $p(\epsilon)/4$ of true. Given m samples, the probability that some member of $\mathcal{K}$ has observed loss differing from true loss by $p(\epsilon)/4$ is (by Hoeffding's inequality) upper bounded by $\exp(-2m(p(\epsilon)/4)^2)$ $= \exp(-mp(\epsilon)^2/8)$.

(Hoeffding's inequality [27] is as follows: Let X_j, $1 \leq j \leq m$ be independent random variables such that $a \leq X_j \leq b$, $1 \leq j \leq m$ for some $-\infty \leq a \leq b \leq \infty$. Then

$$Pr\Big(\frac{1}{m}\sum_{i=1}^{m}[X_i - E(X_i)] \geq \epsilon\Big) \leq \exp\Big[\frac{-2m\epsilon^2}{(b-a)^2}\Big]$$

where we have $a = 0$, $b = 1$.)

So we need $\exp(-mp(\epsilon)^2/8) \leq \delta/|\mathcal{K}|$, *i.e.*

$$\exp(-mp(\epsilon)^2/8) \leq \delta/\Theta((d\log(\delta^{-1})/\zeta^3)^d)$$

$$\Longrightarrow mp(\epsilon)^2/8 \leq \Theta\Big(\log(\delta^{-1}) + d\log(\zeta^{-1}) + d\log(d\log(\delta^{-1}))\Big)$$

$$\Longrightarrow m = \Theta\Big(\frac{1}{p(\epsilon)^2}\Big[\log(\delta^{-1}) + d\log(\zeta^{-1}) + d\log\Big(d\log(\delta^{-1})\Big)\Big]\Big)$$

The above expression for m represents a sufficiently large number of training examples needed, and is a polynomially increasing function. We show below that (for the polynomials p and p' found previously) it is upper bounded by the expression $\Theta(d\log(\delta^{-1})/\zeta^3)$, the number of artificial examples used to construct the ζ-cover (recall that $\zeta = p'^{-1}(\frac{1}{2}p(\epsilon))$). $\square$

Comment: The runtime is polynomial for constant d, and we have shown that the sample complexity is polynomial in d, provided that p and p'^{-1} are polynomial in d. The main computational bottleneck is the generation of a potentially large ζ-cover $\mathcal{K}$ and the measurement of all its elements individually. Under some conditions there may be potential for heuristic elimination from consideration of some elements of $\mathcal{K}$.

Putting it all together, we apply theorem 21 in conjuction with theorem 19. We have

$$p'(\alpha) = 3\alpha, \quad p(\alpha) = \frac{\alpha^8}{2^{44}d^2M(D)^4V(D)^2}$$

Hence $\zeta = (\frac{1}{3})(\frac{1}{2})p(\epsilon) = \epsilon^8/(6 \cdot 2^{44} d^2 M(D)^4 V(D)^2)$. The sample complexity is thus

$$O\Big(\frac{d^7 \log(\delta^{-1}) M(D)^{12} V(D)^6}{\epsilon^{24}}\Big)$$

the size of the set S used to generate the ζ-cover, which dominates the expression for sample size m derived in theorem 21. This is polynomial in δ^{-1} and ϵ^{-1}, and also is polynomial in d for the classes of input distributions identified in examples 8 and 9 (the uniform distribution over the unit hypercube, or normal distributions with unit covariance matrix).

3.3 Conversion to Statistical Queries

The study of PAC-learning in the presence of uniform misclassification noise was introduced in Angluin and Laird [1]. The assumption is that with some fixed probability $\nu < \frac{1}{2}$, any example presented to the learner has had its class label reversed. This is a more realistic model for the data set that motivated this work, in view of the known class overlap. However the algorithm we have presented so far has assumed that the data are noise-free (so that the 1-RFA data came from vectors that are linearly separable). In the presence of noise, the algorithm is not generally guaranteed to converge to the target function. It is shown in [6] how to convert k-RFA learning algorithms to SQ learning algorithms over the boolean domain $\{0,1\}^d$, for k logarithmic in the dimension. Over the real domain not all learning algorithms are amenable to that conversion. We show how to convert our algorithm for linear threshold functions.

The statistical query (SQ) learning framework of Kearns [28] is a restriction of the PAC framework in which the learner has access to unlabeled data, and may make queries of the following form: Any query specifies a predicate χ which takes as input a labeled example (χ should be evaluatable in polynomial time), and an error tolerance α. The response to the query is an estimate of the probability that a random labeled example satisfies χ — the estimate is accurate to within additive error α. The α's used in the queries should be polynomial in the target accuracy ϵ.

Queries of the above form can be answered using a labeled data set in the standard PAC setting. Kearns shows in [28] that they can moreover be answered using a data set with uniform misclassification noise as defined above. If ν_b is a given upper bound on an unknown noise rate ν, then an SQ algorithm would be polynomial in $1/(\frac{1}{2} - \nu_b)$, as well as other parameters of interest (which is how the definition of PAC learning extends to the definition of noise-tolerant PAC learning).

We show how step 3 can be re-cast in the SQ framework. That is, for a given linear threshold function H, use statistical queries to estimate its expected quadratic loss with small additive error α. First note that given any H, there is a probability distribution of quadratic loss of an example generated by D. Suppose we make a histogram approximation by partitioning the range $[0,1]$ of possible values of quadratic loss into $1/\epsilon'$ intervals of length ϵ', for some small $\epsilon' > 0$, and then for each interval, compute the probability that the quadratic loss of a random example lies in that range. We can derive an approximation

to the expected quadratic loss, using this histogram, as the sum over intervals, of the probability of quadratic loss lying in that interval, times the value of the mid-point of that interval. That approximation is within additive error $\epsilon'/2$ of the correct value of expected quadratic loss (since the quadratic loss of any individual point is being approximated by a value within $\epsilon'/2$ of the true value). Call this histogram the "best" histogram for value ϵ'.

For target error ϵ, let $\epsilon' = p(\epsilon)/4$, where p is the polynomial identified in theorem 19, and subsequently used in theorem 21. As in theorem 21, each member H of $\mathcal{K}$ has its expected quadratic losses estimated to within additive error ϵ'. For each interval $\subseteq [0,1]$ of the form $[k\epsilon', (k+1)\epsilon']$ where k is an integer, we make the statistical query: χ is the property that an example has quadratic loss (with respect to H) in the range $[k\epsilon', (k+1)\epsilon']$, and $\alpha = \epsilon'^2/2$. Then the answers to these queries provide an approximation to the best histogram for value ϵ' of the distribution of quadratic loss of labeled examples with respect to H. In particular the histogram found approximates the best to within variation distance $\epsilon'/2$, since each bar of the histogram has its height approximated to within $\epsilon'^2/2$, and there are $1/\epsilon'$ bars. Hence the mean derived from the discovered histogram is within $\epsilon'/2$ of the mean of the best histogram, and consequently within ϵ' of the true mean.

4 The Discrete Boolean Domain

An important special case of the problem is when the input distribution has its domain of support restricted to the boolean domain $\{0,1\}^d$. This restriction affects the learning problem by making it rather trivial for constant d, but apparently still hard if d is not constant. In more detail:

1. The sample complexity is polynomial in the PAC parameters for any fixed d, since the distribution satisfies the conditions of corollary 11. (That result is known from [17].) It is unknown whether the sample complexity is also polynomial in d.

2. There are only $4d$ different observations possible (an observation being the identity of one of the d coordinates together with two possible input values and two possible output values, 0 or 1), so the probability of all of them may be learned with additive error, in time polynomial in d and the reciprocal of the error, by a standard Chernoff bound analysis.

3. For fixed d, there is a fixed number of distinct linear threshold functions, so there is no need for discretization, e.g. via an empirical ϵ-cover.

We note that some knowledge of the input distribution D is still required in this restricted setting. Just three dimensions are needed to allow a pair of indistinguishable scenarios to be constructed.

Fact 22 *It is impossible to learn linear thresholds over the discrete boolean domain* $\{0,1\}^d$ *(for* $d \geq 3$*), if the input distribution is unknown.*

This fact is implied by theorem 3 of [8], since 1-DL, the class of 1-decision lists[2], is a special case of linear threshold functions.

For a given input distribution, the problem is fairly trivial for constant dimensionality d, and in the remainder of this section we discuss the problem for general d.

It is unknown how to efficiently learn perceptrons (linear threshold functions where inputs come from $\{0,1\}^d$) under the uniform input distribution. This is an open problem which predates learning theory, and is in fact the question of how to approximately recover a perceptron from approximations to its *Chow parameters* [17]. The Chow parameters (which are the first-order Fourier coefficients, see [20]) are the set of conditional probabilities that we see in our 1-RFA setting, with D uniform over the boolean domain. It is known from [14, 17] that these parameters do determine the threshold function. As the sample size increases, the $2n$ conditional probabilities will converge to their true values, and it should be possible to reconstruct the coefficients of a suitable linear threshold function given these true values, although even then we do not know how to do so in polynomial time. In any case, it does not follow that it can be done if the observed probabilities have small additive perturbations, as would happen with a finite-sized sample. Indeed it is apparently an open question [21] whether a computationally unbounded learner can be sure to have enough information in a polynomial-sized sample.

A further thing to note about the Chow parameters is that their exact values are hard to compute from exact data. This is due to the fact that the problem of computing them (from some given linear threshold function) is essentially the problem of counting satisfying assignments to the one-dimensional 0/1 knapsack problem, which is $\#P$-hard [22]. (It is in fact open whether one can approximate the number of positive examples on one side of a hyperplane expressed in terms of coefficients and threshold, with small relative error, see [22].) We can however test additively approximate consistency, by random sampling. Note also that our main problem here is finding a (approximate) consistent hypothesis as opposed to testing one.

Regarding the question of what subclasses of perceptrons are 1-RFA learnable, it is known that boolean threshold functions are 1-RFA learnable, for the uniform input distribution. A boolean threshold function is defined by a set of literals and a threshold τ, and evaluates to 1 provided that at least τ of the literals are satisfied. This fact is a special case of the fact from [20] that k-TOP is k-RFA learnable. k-TOP is a class of boolean functions in which instead of monomials we have parity functions over k of the inputs (and then the outputs are input to a threshold gate as in the definition of boolean threshold functions).

[2]A *decision list* is a boolean function defined by a sequence of pairs of monomials and boolean values. A decision list is satisfied by a vector of boolean values whenever the first monomial in the sequence to be satisfied by that vector is paired with the boolean value *true*. For positive integer k, a k-decision list is a decision list whose monomials contain just k literals. Birkendorf et al. [8] show that for $n \geq 3$, $(n-k)$-decision lists are not $(n-k)$-RFA learnable for $k \geq 2$.

5 Conclusion and Open Problems

This paper is the first investigation of restricted focus of attention learning given a known but unrestricted joint distribution of inputs. We have discovered some interesting effects that the joint distribution may have on the number of training examples required for a hypothesis to reach a prescribed level of accuracy. This sensitivity of the sample complexity to the input distribution is evidence of the novelty of the learning situation that we have investigated.

Fundamentally, our algorithm relies on a brute-force approach, which gives the limitation to fixed input dimension d in order to have polynomial runtime. Despite this, it seemed to require fairly sophisticated techniques to obtain the polynomial behavior (in terms of ϵ^{-1} and δ^{-1}). At this stage any improvement in efficiency, allowing the dimensionality to be (for example) logarithmic in the PAC parameters, would be particularly interesting. Since the sample complexity is still polynomial in d for certain classes of input distributions, there may well be possibilities for heuristics to overcome the computational bottleneck. One possibility is elimination of certain members of the unlabeled sample that seem to be nowhere near the threshold.

We suspect that the sufficient conditions for D to give rise to polynomial sample complexity may be extendable much further. So far we have found only the very artificial distributions of section 2.3 which prevent polynomial sample complexity. We conjecture that finite mixtures of distributions that satisfy theorem 10 should be good, even if the domains of different distributions in the mixture have different minimal affine subspaces containing them.

Other open problems include how much knowledge of the input distribution is needed. We know (from fact 22) that even in the boolean domain we do need some knowledge of the input distribution in 3 or more dimensions. If the input distribution D is partly-known, we would like to know to what extent it helps to learn D in the style of [29] if one also has input/output behavior in some given model. One special case of particular interest is when D is known to be a general Gaussian distribution. Then 1-RFA data will not convey information about the covariances, but 1-RFA data labeled by an unspecified linear threshold function might be usable to find covariances. Another question of interest is whether linear threshold functions over the continuous domain can be learned if D is known to be a product distribution, and whether some product distributions make the problem harder than others.

Note that for well-behaved input distributions we would expect to have most difficulty predicting class labels of points near the threshold. We may ask under what circumstances it may be possible to learn in the sense of [9] for learning in situations where points near the boundary may be mislabeled.

For practical purposes we would like to extend these results to deal with the presence of other models of class overlap besides just uniform misclassification noise. The experimental work of [12, 18] assumes members of different classes are generated by separate Gaussian sources, and seeks the best linear threshold (minimum misclassification rate). There are also many possible extensions to other stochastic missing-data mechanisms, which may be

of practical importance while invalidating the general approach presented here. Given the widespread use of imputation as a practical statistical method to deal with missing data, it would be interesting to know whether the PAC criterion for successful learning can ever be achieved by an imputation-based algorithm.

6 Acknowledgements

I would like to thank Mike Paterson for reading an earlier version of this paper and making helpful comments, John Shawe-Taylor, Sally Goldman and Eli Dichterman for helpful information, and David Lowe for initiating my interest in the topic of missing data. My thanks to the referees for their valuable corrections and comments. This work was partially supported by ESPRIT Project ALCOM-IT (Project 20244).

References

[1] D. Angluin and P. Laird (1988). Learning from noisy examples. *Machine Learning* **2**(4), 343-370.

[2] M. Anthony and N. Biggs (1992). *Computational Learning Theory*, Cambridge Tracts in Theoretical Computer Science, Cambridge University Press, Cambridge.

[3] P.L. Bartlett, S.R. Kulkarni and S.E. Posner (1997). Covering numbers for real-valued function classes, *IEEE Transactions on Information Theory* **43**(5), 1721-1724.

[4] E.B. Baum (1990). On learning a union of half spaces. *J. Complexity* **6**(1), pp. 67-101.

[5] S. Ben-David and E. Dichterman (1998). Learning with Restricted Focus of Attention, *J. of Computer and System Sciences*, **56**(3), pp. 277-298. (earlier version in COLT'93)

[6] S. Ben-David and E. Dichterman (1994). Learnability with restricted focus of attention guarantees noise-tolerance, *5th International Workshop on Algorithmic Learning Theory*, pp. 248-259.

[7] S. Ben-David, A. Itai and E. Kushelivitz (1995). Learning by distances, *Information and Computation* **117**(2), 240-250. (earlier version in COLT'90)

[8] A. Birkendorf, E. Dichterman, J. Jackson, N. Klasner and H.U. Simon (1998). On restricted-focus-of-attention learnability of Boolean functions, *Machine Learning*, **30**, 89-123. (earlier version in COLT'96)

[9] A. Blum, P. Chalasani, S.A. Goldman and D.K. Slonim (1998). Learning With Unreliable Boundary Queries, *J. of Computer and System Sciences*, **56**(2) pp. 209-222. (earlier version in COLT'95)

[10] A. Blum, A. Frieze, R. Kannan and S. Vempala (1998). A Polynomial-time Algorithm for Learning Noisy Linear Threshold Functions, to appear in *Algorithmica*; (earlier version in FOCS'96)

[11] A. Blumer, A. Ehrenfeucht, D. Haussler and M.K. Warmuth (1989). Learnability and the Vapnik-Chervonenkis Dimension, J.ACM **36**, 929-965.

[12] J. O'Brien (1998). An Algorithm for the Fusion of Correlated Probabilities. *Procs. of FUSION'98, Las Vegas, July 1998.*

[13] H. Brönnimann and M.T. Goodrich (1994). Almost optimal set covers in finite VC-dimension. *procs. of the 10th Annual Symposium on Computational Geometry* pp. 293-302.

[14] J. Bruck (1990). Harmonic analysis of polynomial threshold functions. *SIAM Journal of Discrete Mathematics*, **3** (2), 168-177.

[15] N.H. Bshouty, P.W. Goldberg, S.A. Goldman and H.D. Mathias (1999). Exact learning of discretized geometric concepts. *SIAM J. Comput.* **28**(2) pp. 674-699.

[16] N.H. Bshouty, S.A. Goldman, H.D. Mathias, S.Suri and H. Tamaki (1998). Noise-Tolerant Distribution-Free Learning of General Geometric Concepts. *Journal of the ACM* **45**(5), pp. 863-890.

[17] C.K. Chow (1961). On the characterisation of threshold functions. *Proc. Symp. on Switching Circuit Theory and Logical Design*, 34-38.

[18] K. Copsey (1998). A Discounting Method for Reducing the Effect of Distribution Variability in Probability-level Data Fusion. *Procs. of EuroFusion'98, Great Malvern, UK, Oct. 1998.*

[19] Dempster, Laird and Rubin (1977). Maximum Likelihood from Incomplete Data via the EM Alorithm, *J. Roy. Stat. Soc.* **39**(1), 1-38.

[20] E. Dichterman (1998). Learning with Limited Visibility. *CDAM Research Reports Series, LSE-CDAM-98-01* 44pp.

[21] E. Dichterman (1999). personal communication.

[22] M. E. Dyer, A. M. Frieze, R. Kannan, A. Kapoor, L. Perkovic and U. Vazirani (1993). A mildly exponential time algorithm for approximating the number of solutions to a multidimensional knapsack problem. *Combinatorics, Probability and Computing* **2**, 271-284.

[23] B. Gärtner and E. Welzl (1996). Linear Programming - Randomization and Abstract Frameworks, *Proc. 13th Annual Symp. on Theoretical Aspects of Computer Science*, LNCS 1064 pp. 669-687.

[24] S.A. Goldman, S.S. Kwek and S.D. Scott (1997). Learning from Examples with Unspecified Attribute Values. *Tenth annual COLT conference*, 231-242, ACM Press, New York.

[25] S.A. Goldman and R.H. Sloan (1995). Can PAC Learning Algorithms Tolerate Random Attribute Noise? *Algorithmica* **14**, 70-84.

[26] D. Haussler (1992). Decision Theoretic Generalizations of the PAC Model for Neural Net and Other Learning Applications, *Information and Computation*, **100**, 78-150.

[27] W. Hoeffding (1963). Probability inequalities for sums of bounded random variables, *J. Amer. Statist. Assoc.* **58**, 13-30.

[28] M.J. Kearns (1993). Efficient Noise-Tolerant Learning From Statistical Queries, *Procs. of the 25th Annual Symposium on the Theory of Computing*, 392-401.

[29] M. Kearns, Y. Mansour, D. Ron, R. Rubinfeld, R.E. Schapire and L. Sellie, On the Learnability of Discrete Distributions, *Proceedings of the 26th Annual ACM Symposium on the Theory of Computing*, (1994) 273–282.

[30] M.J. Kearns and R.E. Schapire (1994). Efficient Distribution-free Learning of Probabilistic Concepts, *Journal of Computer and System Sciences*, **48**(3) 464-497. (earlier version in FOCS '90)

[31] M.J. Kearns and U. Vazirani (1994). *An Introduction to Computational Learning Theory.* MIT Press.

[32] Little and Rubin (1987). *Statistical Analysis with Missing Data*, John Wiley and Sons. ISBN: 0-471-80254-9

[33] D. Lowe and A.R. Webb (1990). Exploiting prior knowledge in network optimization: an illustration from medical prognosis, *Network* **1**, 299-323.

[34] R. Von Mises (1964). *Mathematical Theory of Probability and Statistics.* Academic Press.

[35] D. Pollard (1984). *Convergence of Stochastic Processes*, Springer-Verlag, Berlin/New York.

[36] R.H. Sloan (1988). Types of noise in data for concept learning, *First Workshop on Computational Learning Theory*, 91-96, Morgan-Kaufman.

[37] D.M. Titterington, G.D. Murray, L.S. Murray, D.J. Spiegelhalter, A.M. Skene, J.D.F. Habbena and G.J.Gelpke (1981). Comparison of Discrimination Techniques Applied to a Complex Data Set of Head Injured Patients, *J. Roy. Stat. Soc.* **144**, 145-175.

[38] L.G. Valiant (1984). A Theory of the Learnable. *Commun. ACM* **27**(11), pp. 1134-1142.

[39] L.G. Valiant (1985). Learning disjunctions of conjunctions. *Procs. of 9th International Joint Conference on Artificial Intelligence.*