Learning Noisy Perceptrons by a Perceptron in Polynomial Time

Edith Cohen*
AT&T Labs–Research
Florham Park, NJ 07932
edith@research.att.com
http://www.research.att.com/~edith

September 24, 1997

Abstract

Learning perceptrons (linear threshold functions) from labeled examples is an important problem in machine learning. We consider the problem where labels are subjected to *random classification noise*. The problem was known to be PAC-learnable via a hypothesis that consists of a polynomial number of linear thresholds (due to Blum, Frieze, Kannan, and Vempala). The question whether a hypothesis that is itself a perceptron (a single threshold function) can be found in polynomial time was open. We show that indeed, noisy perceptrons are PAC learnable with a hypothesis that is a perceptron. Our techniques extend to realistic non-uniform noise distributions, where the noise intensity tends to be correlated with proximity to the optimal linear threshold.

A different phrasing of our main result is as follows. Let χ be a distribution on R^n contained in a halfspace $\{\mathbf{x} \in R^n | \mathbf{x} \cdot \mathbf{h} > 0\}$. χ is subjected to noise level $\Delta < 0.5$, where the sign of each sample $\mathbf{x} \in \chi$ is flipped with probability Δ. We present a polynomial time algorithm (in ϵ^{-1}, n, $(1-2\Delta)^{-1}$ and the size of the binary representation of the samples) that accesses random samples from the noisy distribution and outputs a vector $\mathbf{g} \in R^n$ such that with high confidence, $\mathrm{Prob}_{\mathbf{x}\in\chi}\{\mathbf{g} \cdot \mathbf{x} \leq 0\} < \epsilon$.

1. Introduction

The *perceptron problem* (learning a linear threshold function) is one of the oldest problems studied in machine learning. Linear threshold functions (perceptrons) are well-studied representation for learning from examples whose usefulness has been demonstrated on many

*Part of this research was done while the author was visiting U.C. Berkeley and supported by NSF grant CCR-9616139

datasets. Perceptrons generalize many representation studied in the computational learning theory literature, for example, disjunctions of literals, boolean threshold functions [13], and decision lists [15].

The noise-free formulation of the problem is as follows. Consider two probability distributions over R^n, one of "positive" and one of "negative" examples such that there exists a hyperplane (linear threshold function) that separates the regions where the distributions have nonzero probability. Given access to labeled examples (random samples), the goal is to find a linear threshold function that classifies correctly $(1-\epsilon)$ of the examples. A straightforward transformation reduces the problem to such where the desired hyperplane goes through the origin (replace each example $\mathbf{x} = (x_1, \ldots, x_n) \in R^n$ by $(x_1, \ldots, x_n, 1) \in R^{n+1}$). By negating all negatively-labeled vectors (see [9]), we obtain the following equivalent formulation. Let χ be a distribution on R^n such that there exist a halfspace H such that χ has zero measure outside H. Suppose we are given access to random samples from χ. The objective is to identify, with very high confidence, a vector $\mathbf{g}$ such that $\mathbf{x} \cdot \mathbf{g} > 0$ for at least $(1-\epsilon)$ of the vectors $\mathbf{x} \in \chi$. This noise-free perceptron problem is known to be solvable in polynomial time using any Linear Programming algorithm. One of the oldest algorithms for this problem is the *perceptron algorithm* [16, 1], which is still commonly-used and has a good empirical behaviour [3]. The algorithm converges to a good solution, but the running time depends on a *wiggle room* parameter which could be exponentially small in the size of the binary representation of the examples. Hence, the perceptron algorithm does not have a polynomial running time.

This paper focuses on the noisy version of the perceptron problem with *random classification noise.* A noise level of $0 < \Delta < 0.5$ means that the label of each example (positive or negative) is flipped (equivalently, the sign of a vector sampled from χ is negated) with probability Δ before being presented to us. The objective here is to identify a halfspace that contains at least $(1-\epsilon)$ of the *original* distribution. Our main result establishes that this problem is PAC learnable. (See [11] for background on PAC learning. Knowledge of these terms in not necessary for reading this paper.)

Random classification noise, which implies uniform noise-level over all data, may not always be a realistic assumption. Our results extend to data with some non-uniform (and unknown) noise distributions, when the noise level is more highly correlated with proximity to optimal linear thresholds than with "unsatisfactory" linear thresholds.

An equivalent formulation of the noisy perceptron problem is as follows: Let χ be a distribution in R^n such that there exist a vector $\mathbf{h}$ such that for any $\mathbf{x} \in \chi$, $\mathbf{h} \cdot \mathbf{x} > 0$. The distribution χ is subject to noise, resulting in a new distribution $\tilde{\chi}$. Let $0 < \Delta < 0.5$ be the amount of random noise. $\tilde{\chi}$ is generated from χ by negating vectors with probability Δ. That is, a random sample $\tilde{\mathbf{x}} \in \tilde{\chi}$ can be seen as drawing a sample $\mathbf{x} \in \chi$ and then outputting

$$\tilde{\mathbf{x}} = \begin{cases} \mathbf{x} & \text{with probability } 1-\Delta \\ -\mathbf{x} & \text{with probability } \Delta. \end{cases}$$

We are given access to samples from $\tilde{\chi}$. The objective is to find a vector $\mathbf{g}$ such that $\mathrm{Prob}_\chi\{\mathbf{x} \cdot \mathbf{g} \leq 0\} < \epsilon$. Our main result is showing that for any χ, Δ, and ϵ, this can be done with very high probability $(1 - e^{-poly(b,n,\epsilon^{-1},(1-2\Delta)^{-1})})$ in time polynomial in b, n, ϵ^{-1}, and $(1-2\Delta)^{-1}$, where b is the maximum number of bits in a binary representation of the sampled vectors.

Some notable previous work on the problem is a result by Kearns [10] who established PAC learnability for some class of distributions (radially symmetric), by showing that the problem falls in his statistical query model. Bylander [5] provided a modification of the (non polynomial) perceptron algorithm that under some additional assumptions, converges in the presence of random classification noise. Recently, he extended his algorithm to "probabilistically consistent linear threshold functions." The latter refers to noisy data where the noise is non-decreasing with proximity to the optimal threshold [6]. A recent major step was taken by, Blum, Frieze, Kannan, and Vempala [4]. They showed that noisy linear threshold functions (with random classification noise) can be learned (in the PAC model sense) in polynomial time via a hypothesis that consists of a *polynomial number* of linear thresholds (hyperplanes). Their hypothesis tests a sample point sequentially against a polynomial number of linear thresholds until one that agrees with the point is encountered (and the point gets a positive label) or none does (negative label). The Blum *et al.* [4] algorithm utilizes Bylander's modified perceptron algorithm repeatedly in many iterations. Each iteration runs in polynomial time and yields another condition to the classification rule that classifies well a polynomial fraction of the remaining unclassified examples. Blum *et al.* posed as an open problem the question whether, as is the case for the noise-free version, noisy perceptrons (linear threshold functions subject to random classification noise) are PAC learnable via a hypothesis that consists of *a single threshold function.* (Can a single hyperplane that separates at least $1-\epsilon$ of the distribution be identified in polynomial time?) Note that such a hyperplane exists by the definition of the problem. Polynomial time learning of noisy perceptrons by a *single* threshold function has both theoretical and practical significance. We answer their question on the affirmative.

The noise-free perceptron problem is known to be polynomial-time solvable using any Linear Programming algorithm. The algorithm simply converts sample points $\mathbf{x} \in \chi$ to constraints $\mathbf{x} \cdot \mathbf{g} > 0$ and solves the resulting linear system. VC dimension argument establishes that a polynomial number $(O(\epsilon^{-1} \log \delta^{-1} + n\epsilon^{-1} \log \epsilon^{-1}))$ of sample points is very likely to yield a solution (probability $1-\delta$). This LP approach does not seem to generalize to the noisy problem. In the presence of noise, the system obtained as described above is generally infeasible. The objective is to identify a vector $\mathbf{g}$ consistent with "most" of the constraints. The difficulty stems from the fact that it is NP-hard to find a solution to an infeasible linear system that minimizes the number of unsatisfied constraints. The problem remains NP-hard under some relaxations when an exact minimum is not required. Our algorithm utilizes Linear Programming via a different approach. We apply the Ellipsoid algorithm to the (non-explicit) system formed by the non-noisy constraints. The Ellipsoid algorithm solves such systems with the aid of a *separation oracle* that produces violated constraints in response to inappropriate classifiers (infeasible points). The main challenge in our work was to implement a "separation oracle" which is correct with very high probability. The main ingredients we used were leveling down the effects of noise by "averaging", restricting the distribution to "slices" of the space, and using the *outliers removal lemma* of [4]. Section 2 provides preliminaries and an outline of our algorithm. Section 3 is concerned with the "separation oracle". Section 4 discusses our use of the Ellipsoid algorithm. In Section 5 we extend our results to some non-uniform noise distributions. Section 6 summarizes and relates our and previous algorithmic ideas. In Section 7 we explore future research directions.

2. Preliminaries and Outline

By *high confidence* or *high probability* we mean probability at least $1-e^{-\mathrm{poly}_1(n,b,\epsilon^{-1},(1-2\Delta)^{-1})}$, for any poly_1 of our choice. By *polynomial* (for the time or number of samples) we mean: no larger than some $\mathrm{poly}_2(n,b,\epsilon^{-1},(1-2\Delta)^{-1})$. Conveniently, a polynomial number of polynomial-time high-probability events is also a polynomial-time high-probability event.

For notational convenience we interchange between the distribution being defined on the n dimensional sphere S^n or on the n-dimensional space R^n. When considering a vector we sometimes interchange between the vector itself and the normalized version. The correct meaning should be clear from context.

For the sake of clean and clear presentation, we make several simplifying assumptions that do not affect generality. We assume that the noise level $\Delta < 0.5$ is known. This assumption can be easily removed using standard arguments [11] or by modifying the algorithm. We assume that χ has zero measure on any one hyperplane going through the origin, that is, for any $\mathbf{u} \neq \mathbf{0}$, $\mathrm{Prob}_{\mathbf{x}\in\chi}\{\mathbf{x}\cdot\mathbf{u} = 0\} = 0$. This clearly does not hold when χ is discrete, which is the case here since we assume that vector are represented using at most b bits. Our arguments, however, can be easily modified to handle the general case. Alternatively, small random perturbation of the data can force the condition $\mathrm{Prob}_{\mathbf{x}\in\chi}\{\mathbf{x}\cdot\mathbf{u} = 0\} \leq 2^{-\mathrm{poly}}$ (also by increasing the vector size to the appropriate poly) which suffices for our arguments. We required that *all* samples from χ are of size at most b. This can be clearly generalized to requiring that vectors in $(1-(1+1/\mathrm{poly})\epsilon)$ fraction of χ have size at most b.

We used *strict* inequalities in the perceptron problem definition, that is, we required that there exist an *open halfspace* that contains the distribution χ. This assumption simplifies some of our arguments, and in particular, guarantees that the "solution space" is *fully dimensional*, that is, the polyhedron $F' = \{\mathbf{h} \in R^n | \forall \mathbf{x} \in \chi, \mathbf{h}\cdot\mathbf{x} > 0\}$ has dimension n. We note, however, that our approach is applicable to non-strict inequalities using standard arguments in the optimization literature (see, e.g., [8]).

We utilize the following additive version of Chernoff bound [7] throughout our analysis.

Proposition 2.1 [Chernoff Bound] *Consider N independent and identically distributed random variables $x_1,\ldots,x_N$ defined on the interval $[0,1]$ with expectation μ. For any $\eta > 0$ we have*

$$\mathrm{Prob}\left\{\frac{\sum_{i=1}^N x_i}{N} > \mu+\eta\right\} \leq e^{-2N\eta^2}$$
$$\mathrm{Prob}\left\{\frac{\sum_{i=1}^N x_i}{N} < \mu-\eta\right\} \leq e^{-2N\eta^2}$$

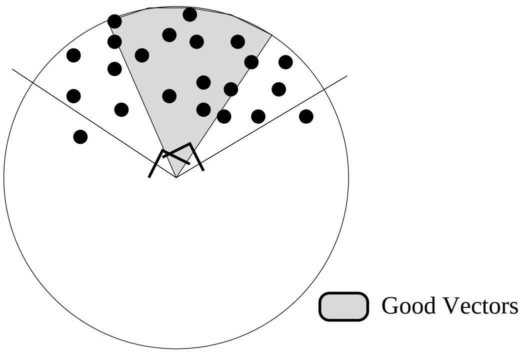

Figure 1: Visualizing good vectors ($\epsilon \leq 1/22$)

2.1. Good and Bad Vectors

Definition 2.2 A vector $\mathbf{g}$ is *Good* if

$$\operatorname*{Prob}_{\chi}\{\mathbf{x} \cdot \mathbf{g} < 0\} < \epsilon$$

or equivalently,

$$\operatorname*{Prob}_{\tilde{\chi}}\{\mathbf{x} \cdot \mathbf{g} < 0\} < \Delta + \epsilon(1 - 2\Delta) \ .$$

(since $\Delta(1-\epsilon)+(1-\Delta)\epsilon = \Delta+\epsilon(1-2\Delta)$.) Figure 1 visualizes good vectors for a distribution χ on R^2. The black circles correspond to samples from χ. The good vectors are all those that exclude at most one sample. Our objective is to identify a good vector. Naturally, *bad vectors* are vectors that are not good. However, we relax the definition slightly and allow a grey area: a vector $\mathbf{v}$ is *bad* if

$$\operatorname*{Prob}_{\chi}\{\mathbf{x} \cdot \mathbf{v} < 0\} > \epsilon/2$$

or equivalently,

$$\operatorname*{Prob}_{\tilde{\chi}}\{\mathbf{x} \cdot \mathbf{v} < 0\} > \Delta + \frac{\epsilon(1 - 2\Delta)}{2} \ .$$

The grey area is introduced to absorb estimation inaccuracies.

Badness or goodness of vectors can be identified with high probability using polynomially many samples. Vectors which are not good are identified as bad and vectors which are not bad are identified as good. "Grey" vectors are identified either way.

Proposition 2.3 *For any vector* $\mathbf{v} \in R^n$*, badness or goodness can be identified using* r *samples from* $\tilde{\chi}$ *with confidence at least* $1 - e^{-\Omega(r(\epsilon(1-2\Delta))^2)}$.

Proof: Let S be a set of r samples from $\tilde{\chi}$. Let $S^- = \{\mathbf{x} \in S | \mathbf{x} \cdot \mathbf{v} < 0\}$. and define $f = \frac{|S^-|}{|S|}$. If $f \leq \Delta + \frac{3}{4}\epsilon(1-2\Delta)$ we identify $\mathbf{v}$ as good and otherwise as bad. The identification is wrong if f is more than $\frac{1}{4}\epsilon(1-2\Delta)$ off its expected value. From Chernoff bounds (Proposition 2.1) it follows the probability of wrong identification is at most

$$2e^{-2r(\epsilon(1-2\Delta)/4)^2} = 2e^{-r(\epsilon(1-2\Delta))^2/8} .$$

∎

2.2. Outliers Removal

The Outlier Removal Lemma of Blum *et al.* [4] states that for any distribution χ on R^n and $\epsilon' \geq 2^{-nb}$, we can remove ϵ' fraction of the distribution χ to obtain χ' such that for all $\mathbf{u} \in R^n$

$$\max_{\mathbf{x}\in\chi'} |\mathbf{x}\cdot\mathbf{u}| \leq \beta \mathop{\mathrm{E}}_{\mathbf{x}\in\chi'} |\mathbf{x}\cdot\mathbf{u}| .$$

Where $\beta = \mathrm{poly}(n, b, \epsilon'^{-1})$. The removal procedure identifies a polynomial number of ellipsoids such that a sample point $\mathbf{x}$ is to be removed if it is not a member of all of the ellipsoids. The construction of the ellipsoids is performed in time $\mathrm{poly}(n, b, \epsilon'^{-1})$. The removal procedure imposes a $\mathrm{poly}(n, b, \epsilon'^{-1})$ time overhead on obtaining each sample from $\tilde{\chi}$.

Our algorithm utilizes outliers removal. The algorithm restricts the distribution to *slices* that are symmetric subsets of R^n. We need outliers removed both from the original distribution χ or from χ restricted to a slice. We always consider slices one at a time and do not select them recursively. Therefore, outliers can be removed when a slice is considered, with respect to the current slice alone.

For cleanliness of presentation, we assume from here onward that outliers removal is done transparently. That is, when a slice of χ is considered, we assume that outliers are "automatically" removed. The loss of a fraction of the distribution can be absorbed by adjusting the allowed ϵ by a constant factor. For example, if the objective is to identify a good vector with respect to ϵ, we first remove outliers with $\epsilon' = \epsilon/2$ and then find a good vector $\mathbf{g}$ on the new distribution for $\epsilon/2$. It is easy to see that $\mathbf{g}$ is a good vector on the original distribution with respect to ϵ. From here on, we do not explicitly bother with accounting for ϵ's. This holds as long as $\epsilon = \Omega(2^{-nb})$. (The latter is not really a restriction since otherwise we allow for exponential time in the problem size.)

2.3. The Feasible Region

The material in this section is needed for the analysis only. (See Figure 2 for a 2-dimensional example.)

Proposition 2.4 *There exist a set of $k = n$ vectors $\{\mathbf{h}_1, \ldots \mathbf{h}_k\} \subset S^n$ such that:*

1. *For $i = 1, \ldots, k$, $\forall \mathbf{x} \in \chi$ $\mathbf{h}_i \cdot \mathbf{x} > 0$.*

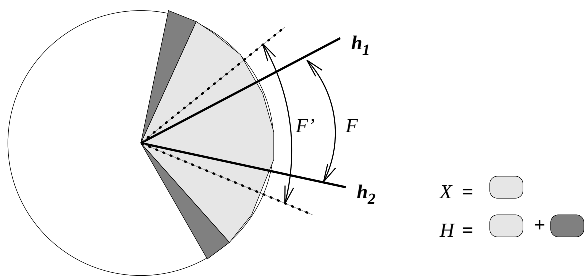

Figure 2: 2-dim example of X, H, F, F', and $\{\mathbf{h}_1, \mathbf{h}_2\}$.

2. *Let* $F = B^n \cap \text{cone}(\mathbf{h}_1, \ldots, \mathbf{h}_k)$, *where* B^n *is the* n*-dimensional unit ball. We have* $\text{vol}(F) \geq 2^{-poly(b,n)}$.

Proof: The proof of Proposition 2.4 uses standard polyhedral theory (see for example [8]). Let $X \subset R^n$ be cone(χ), that is, X contains all nonnegative linear combinations of vectors in χ. Let

$$F' = \{\mathbf{h} \in S^n | \forall \mathbf{x} \in \chi, \mathbf{h} \cdot \mathbf{x} > 0\} = \{\mathbf{h} \in S^n | \forall \mathbf{x} \in X, \mathbf{h} \cdot \mathbf{x} > 0\} ,$$

be the feasibility region of the perceptron problem on χ. By definition, F' contains all normalized vectors that completely separate χ. In particular, it is contained in the set of normalized good vectors.

It follows from our assumptions that X is a cone with a null lineality space. (It does not contain any non trivial subspace). Hence, standard polyhedral theory establishes that cone(F') is full dimensional. Each edge of cone(F') is a solution of $n-1$ linear equations of the form: find $\mathbf{y}$ such that $\mathbf{y} \cdot \mathbf{x}_i = 0$ for some $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \subset \chi$. Samples from χ have size at most b. Hence, the edge $\mathbf{y} = \{\alpha \mathbf{u} | \alpha > 0\}$ is defined by $\mathbf{u}$ of size (binary representation) at most poly(n, b). Consider n affinely independent edges defined by $\mathbf{u}_1, \ldots, \mathbf{u}_n$. Standard arguments [8] establish that $B^n \cap \text{cone}(\mathbf{u}_1, \ldots, \mathbf{u}_n)$ has volume at least $2^{-\text{poly}(b,n)}$. Moreover, the interior of cone($\mathbf{u}_1, \ldots, \mathbf{u}_n$) is contained in F'. The vectors $\mathbf{u}_i$ $(1 = 1, \ldots, n)$ are not in F', and hence we can not simply choose $\mathbf{h}_i \equiv \mathbf{u}_i$. We choose instead, for $i = 1, \ldots, n$, $\mathbf{h}_i$ to be a slight perturbation of $\mathbf{u}_i$ that is in the interior of F'. It is easy to see that there exist a perturbation such that $vol(B^n \cap \text{cone}(\mathbf{h}_1, \ldots, \mathbf{h}_n))$ is at least $2^{-\text{poly}(b,n)}$. ∎

In Section 3 we establish that some properties hold with high confidence for the vectors $\mathbf{h}_1, \ldots, \mathbf{h}_k$. These vectors are not provided explicitly. Note that it suffices to show that the property holds with high confidence for one generic $\mathbf{h} \in \{\mathbf{h}_1, \ldots, \mathbf{h}_k\}$. This implies that the property holds with high confidence for any fixed collection of $k = O(2^{\text{poly}})$ such vectors. Hence, we prove claims with respect to a single $\mathbf{h}$.

Denote $H = \{\mathbf{x} \in R^n | \mathbf{h}_i \cdot \mathbf{x} > 0 \text{ for } i = 1, \ldots, k\}$. The vectors $\mathbf{h}_1, \ldots, \mathbf{h}_k$, a generic $\mathbf{h}$, and the sets F and H are referred to later on in establishing correctness and convergence of our algorithm.

2.4. Sketch of our algorithm

Definition 2.5 A vector $\mathbf{w}$ is a *witness* for a bad vector $\mathbf{v}$ if

1. $\mathbf{w}$ is in H ($\mathbf{w} \cdot \mathbf{h}_i > 0$ for $i = 1, \ldots k$). (see Section 2.3).

2. $\mathbf{w}$ has a negative dot product with $\mathbf{v}$ ($\mathbf{w} \cdot \mathbf{v} < 0$).

The following theorem, which is proved in Section 3, is central to our algorithm:

Theorem 2.6 Witness Identification: *For any bad vector $\mathbf{v} \in R^n$, a witness vector $\mathbf{w}$ can be found in polynomial time with very high confidence*

In Section 4 we employ the *Ellipsoid Algorithm* [12] to find a good vector with Theorem 2.6 serving as the *separation oracle*. The Ellipsoid algorithm is applied to the following (very large and non-explicit) linear system of inequalities:

$$\text{find a vector } \mathbf{g} \in R^n \text{ such that:} \quad \text{for all } \mathbf{x} \in H,\ \mathbf{g} \cdot \mathbf{x} > 0 \ .$$

In each iteration the Ellipsoid algorithm considers a vector $\mathbf{v} \in R^n$ (center of current ellipsoid). The vector $\mathbf{v}$ is tested for goodness. If $\mathbf{v}$ is good, it constitutes a solution and we stop. If $\mathbf{v}$ is bad, then Theorem 2.6 finds a witness $\mathbf{w}$. The inequality $\mathbf{x} \cdot \mathbf{w} < 0$ constitutes a constraint violated by $\mathbf{v}$ and is presented to the Ellipsoid algorithm.

It is easy to see that if $\mathbf{v}$ is bad then $\epsilon(1 - 2\Delta)/2$ fraction of the samples $\mathbf{x} \in \tilde{\chi}$ qualify as witnesses. The challenge in proving Theorem 2.6 is in identifying a vector that with *very high confidence* is a witness. Property 2 of a witness can be verified directly. The difficulty is in verifying Property 1, since the vectors $\mathbf{h}_i$ (or any other feasible vectors) are not known. A confidence of at least $1 - 1/\text{poly}$ in identifying witnesses is necessary for a good success probability of the Ellipsoid algorithm. The algorithm requires correct witnesses in *all* (polynomially many) iterations.

3. Proof of the Witness Identification Theorem

We present an algorithm that for any bad vector $\mathbf{v}$ finds a witness $\mathbf{w}$ as promised in Theorem 2.6. Note that to establish Property 1 of a witness, it is sufficient to show that it holds with high confidence for a generic $\mathbf{h} \in \{\mathbf{h}_1, \ldots, \mathbf{h}_k\}$.

We sketch and motivate our approach. Consider samples $\mathbf{x} \in \tilde{\chi}$ such that $\mathbf{v} \cdot \mathbf{x} < 0$. Even though $\frac{\epsilon(1-2\Delta)}{2\Delta + \epsilon(1-2\Delta)}$ fraction of these samples are in χ and hence in H, and therefore, qualify as witnesses, we can not identify with high enough confidence even one such sample. What we can do, fortunately, is consider *centers of mass*. The centers of mass of all samples $\mathbf{x} \in \tilde{\chi}$ where $\mathbf{x} \cdot \mathbf{v} < 0$ and of all samples $\mathbf{x} \in \tilde{\chi}$ where $\mathbf{x} \cdot \mathbf{v} > 0$ can be used to obtain an unbiased estimate of the center of mass $\mathbf{m}^-$ of all $\mathbf{x} \in \chi$ where $\mathbf{x} \cdot \mathbf{v} < 0$. The vector $\mathbf{m}^-$ is a witness: By definition it is in H and has a negative dot product with $\mathbf{v}$. The impediment is that $\mathbf{m}^-$ can not be estimated accurately enough from polynomial number of samples to guarantee the witness properties. We use two key ideas to overcome that. The first is based on the fact that after removing outliers, for $\nu \geq 1/\text{poly}$, $\mathbf{w} = (1 - \nu)\mathbf{m}^- + \nu \mathbf{m}^+$, ($\mathbf{m}^+$ is the center of mass of all $\mathbf{x} \in \chi$ where $\mathbf{x} \cdot \mathbf{v} < 0$) for $\nu \geq 1/\text{poly}$, can we estimated sufficiently accurately with a polynomial number of samples such that with very high probability $\hat{\mathbf{w}} \cdot \mathbf{h} > 0$. Hence, we are assured that witness property 1 holds. The caveat, however, is that $\mathbf{m}^+ \cdot \mathbf{v}$ may be exponentially larger than $|\mathbf{m}^- \cdot \mathbf{v}|$, and therefore, there is no good choice of ν for which $\mathbf{w} \cdot \mathbf{v} < 0$ (witness property 2 is violated). To overcome this, we use our second key idea. Observe that if all $\mathbf{x} \in \chi$ have approximately the same values of $|\mathbf{x} \cdot \mathbf{v}|$, we have $\mathbf{m}^+ \cdot \mathbf{v} = \Omega(\epsilon^{-1} \mathbf{m}^- \cdot \mathbf{v})$ and the "bad case" can not happen. To achieve that, we restrict χ to a sufficiently dense (relative to χ) and "narrow" (relative to $\mathbf{v}$) symmetric regions. *Symmetric regions* are such that if $\mathbf{z}$ is in the region, so is $-\mathbf{z}$. We use symmetric regions of the form $\{\mathbf{x} \in R^n | a_2 > |\mathbf{v} \cdot \mathbf{x}| > a_1\}$ and refer to them as *slices*. An alternative solution we discuss later on is to scale all vectors such that $|\mathbf{x} \cdot \mathbf{v}| = 1$.

3.1. Definitions and Properties

Let ψ be a distribution over the sphere S^n and $\mathbf{u} \in R^n$ be a vector. We define

$$
\begin{aligned}
H^-(\mathbf{u}) &= \{\mathbf{x} \in R^n | \mathbf{u} \cdot \mathbf{x} < 0\} \\
H^+(\mathbf{u}) &= \{\mathbf{x} \in R^n | \mathbf{u} \cdot \mathbf{x} > 0\} \\
M^+(\psi, \mathbf{u}) &= \int_{\mathbf{x} \in H^+(\mathbf{u})} \psi(\mathbf{x}) d\mathbf{x} \\
M^-(\psi, \mathbf{u}) &= \int_{\mathbf{x} \in H^-(\mathbf{u})} \psi(\mathbf{x}) d\mathbf{x} \\
\mathbf{m}^+(\psi, \mathbf{u}) &= \int_{\mathbf{x} \in H^+(\mathbf{u}) \cap S^n} \mathbf{x} \psi(\mathbf{x}) d\mathbf{x} \\
\mathbf{m}^-(\psi, \mathbf{u}) &= \int_{\mathbf{x} \in H^-(\mathbf{u}) \cap S^n} \mathbf{x} \psi(\mathbf{x}) d\mathbf{x}
\end{aligned}
$$

In words: $M^\pm(\psi, \mathbf{u})$ is the measure of ψ on $H^\pm(\mathbf{u})$, and the vectors $\mathbf{m}^\pm$ are the centers of mass of the part of the distribution on $H^\pm(\mathbf{u})$. Our assumptions imply

$$M^+(\chi, \mathbf{v}) + M^-(\chi, \mathbf{v}) = M^+(\tilde{\chi}, \mathbf{v}) + M^-(\tilde{\chi}, \mathbf{v}) = 1 \ .$$

To motivate these definitions, notice that if $\mathbf{v}$ is a bad vector then $\mathbf{m}^-(\chi, \mathbf{v}) \cdot \mathbf{h} > 0$ and $\mathbf{m}^-(\chi, \mathbf{v}) \cdot \mathbf{v} < 0$. Hence, $\mathbf{m}^-(\chi, \mathbf{v})$ is a witness.

The quantities $M^\pm(\tilde{\chi}, \mathbf{v})$ and $\mathbf{m}^\pm(\tilde{\chi}, \mathbf{v})$ can be estimated by sampling from $\tilde{\chi}$. The quantities $M^\pm(\chi, \mathbf{v})$ and $\mathbf{m}^\pm(\chi, \mathbf{v})$ are related to them by the following linear equations:

$$
\begin{aligned}
M^+(\tilde{\chi}, \mathbf{v}) &= (1-\Delta) M^+(\chi, \mathbf{v}) + \Delta M^-(\chi, \mathbf{v}) && (1) \\
M^-(\tilde{\chi}, \mathbf{v}) &= (1-\Delta) M^-(\chi, \mathbf{v}) + \Delta M^+(\chi, \mathbf{v}) && (2) \\
\mathbf{m}^+(\tilde{\chi}, \mathbf{v}) &= (1-\Delta) \mathbf{m}^+(\chi, \mathbf{v}) - \Delta \mathbf{m}^-(\chi, \mathbf{v}) && (3) \\
\mathbf{m}^-(\tilde{\chi}, \mathbf{v}) &= (1-\Delta) \mathbf{m}^-(\chi, \mathbf{v}) - \Delta \mathbf{m}^+(\chi, \mathbf{v}) && (4)
\end{aligned}
$$

Solving the linear systems in Equ. 1–2 and Equ. 3–4 gives:

$$M^+(\chi, \mathbf{v}) = \frac{(1-\Delta) M^+(\tilde{\chi}, \mathbf{v}) - \Delta M^-(\tilde{\chi}, \mathbf{v})}{1 - 2\Delta} \quad (5)$$

$$M^-(\chi,\mathbf{v}) = \frac{(1-\Delta)M^-(\tilde{\chi},\mathbf{v}) - \Delta M^+(\tilde{\chi},\mathbf{v})}{1-2\Delta} \tag{6}$$

$$\mathbf{m}^+(\chi,\mathbf{v}) = \frac{(1-\Delta)\mathbf{m}^+(\tilde{\chi},\mathbf{v}) + \Delta \mathbf{m}^-(\tilde{\chi},\mathbf{v})}{1-2\Delta} \tag{7}$$

$$\mathbf{m}^-(\chi,\mathbf{v}) = \frac{(1-\Delta)\mathbf{m}^-(\tilde{\chi},\mathbf{v}) + \Delta \mathbf{m}^+(\tilde{\chi},\mathbf{v})}{1-2\Delta} \tag{8}$$

3.2. Estimating $M^\pm(\tilde{\chi},\mathbf{v})$ and $\mathbf{m}^\pm(\tilde{\chi},\mathbf{v})$ from samples

Let $S = \{\mathbf{x}_1,\ldots,\mathbf{x}_N\} \subset \chi$ be N (normalized) random samples from $\tilde{\chi}$. (the magnitude of N is polynomial and will be discussed later). Let $S^+ = S \cap H^+(\mathbf{v}) = \{\mathbf{x} \in S | \mathbf{x}\cdot\mathbf{v} > 0\}$ be all samples with positive dot products with $\mathbf{v}$ and similarly, let $S^- = S \cap H^-(\mathbf{v}) = \{\mathbf{x} \in S | \mathbf{x}\cdot\mathbf{v} < 0\}$.

$$\hat{M}^+(\tilde{\chi},\mathbf{v}) = \frac{|S^+|}{|S|} \tag{9}$$

$$\hat{M}^-(\tilde{\chi},\mathbf{v}) = \frac{|S^-|}{|S|} \tag{10}$$

$$\hat{\mathbf{m}}^+(\tilde{\chi},\mathbf{v}) = \frac{\sum_{\mathbf{x}\in S^+}\mathbf{x}}{|S|} \tag{11}$$

$$\hat{\mathbf{m}}^-(\tilde{\chi},\mathbf{v}) = \frac{\sum_{\mathbf{x}\in S^-}\mathbf{x}}{|S|} \tag{12}$$

Estimates $\hat{\mathbf{m}}^\pm(\chi,\mathbf{v})$ and $\hat{M}^\pm(\chi,\mathbf{v})$ can be computed from $\hat{\mathbf{m}}^\pm(\tilde{\chi},\mathbf{v})$ and $\hat{M}^\pm(\tilde{\chi},\mathbf{v})$ in the obvious manner according to Equ. 5–8.

Shorthands. When χ and $\mathbf{v}$ are clear from the context, we use the notation $M^+ \equiv M^+(\chi,\mathbf{v})$, $\mathbf{m}^+ \equiv \mathbf{m}^+(\chi,\mathbf{v})$, $M^+_\sim \equiv M^+(\tilde{\chi},\mathbf{v})$, and $\mathbf{m}^+_\sim \equiv \mathbf{m}^+(\tilde{\chi},\mathbf{v})$. The quantities M^-, $\mathbf{m}^-$, $M^-_\sim$, and $\mathbf{m}^-_\sim$, are defined similarly. For the estimates (Equ. 9–12) we use $\hat{\mathbf{m}}^+_\sim \equiv \hat{\mathbf{m}}^+(\tilde{\chi},\mathbf{v})$, $\hat{\mathbf{m}}^-_\sim \equiv \hat{\mathbf{m}}^-(\tilde{\chi},\mathbf{v})$, and similarly define $\hat{M}^\pm_\sim$ and $\hat{m}^\pm$.

3.3. Accuracy of Estimates

Consider a distribution ψ on S^n and an arbitrary vector $\mathbf{u} \in S^n$. We apply Chernoff bounds to bound the accuracy and confidence of estimating from samples the expected magnitudes of projections on $\mathbf{u}$.

Proposition 3.1 *Let ψ be a distribution over the sphere S^n and let $\mathbf{u} \in S^n$ be a vector. Suppose that outliers were removed with respect to $\mathbf{u}$, that is,*

$$\max_{\mathbf{x}\in\psi} |\mathbf{x}\cdot\mathbf{u}| \le \beta \mathop{\mathrm{E}}_{\mathbf{x}\in\psi}\{|\mathbf{x}\cdot\mathbf{u}|\} \ .$$

Let $\mathbf{a} = \mathrm{E}_{\mathbf{x}\in\psi}\{\mathbf{x}\}$ *be the center of mass of* ψ*. Consider a set* S *of* $r = |S|$ *samples drawn from* ψ*. Let*

$$\hat{\mathbf{a}}_r = \frac{\sum_{\mathbf{x}\in S}\mathbf{x}}{|S|}$$

be an estimate of $\mathbf{a}$*. For all* $\gamma > 0$*,*

$$|(\mathbf{a} - \hat{\mathbf{a}}_r)\cdot\mathbf{u}| \le \gamma \mathop{\mathrm{E}}_{\mathbf{x}\in\psi}\{|\mathbf{x}\cdot\mathbf{u}|\} \ ,$$

with confidence at least $1 - e^{-\Omega(r\frac{\gamma^2}{\beta^2}))}$*.*

Proof: The random variable $\frac{\mathbf{x}\cdot\mathbf{u}}{\max_{\mathbf{x}\in\psi}|\mathbf{x}\cdot\mathbf{u}|}$ has values in the range $[-1,1]$ and we need to estimate its expectation to within $\pm\gamma/\beta$. This is equivalent to estimating the expectation of an r.v. in $[0,1]$ to within $\eta = \frac{\gamma}{2\beta}$. From Chernoff bounds (see Proposition 2.1), r samples provide confidence of

$$1 - 2e^{-2r\eta^2} = 1 - 2e^{-r\frac{\gamma^2}{2\beta^2}} \ .$$

∎

Consider the noisy distribution $\tilde{\chi}$ and a vector $\mathbf{v}$. Let S be a set of samples from $\tilde{\chi}$ as in Section 3.2.

Let $\mathbf{u} \in R^n$ be an arbitrary vector.

Proposition 3.2 *If* $|S| = N$ *samples are used then for any* $\gamma > 0$*, with probability at least* $1 - e^{-\Omega(\frac{NM^+_\sim\gamma^2}{\beta^2})}$*,*

$$|(\hat{\mathbf{m}}^+_\sim - \mathbf{m}^+_\sim)\cdot\mathbf{u}| \le \gamma \mathop{\mathrm{E}}_{\mathbf{x}\in\tilde{\chi}\cap H^+(\mathbf{v})}\{|\mathbf{u}\cdot\mathbf{x}|\}M^+_\sim \ .$$

The symmetric claims hold for $\hat{\mathbf{m}}^-_\sim$ *estimating* $\mathbf{m}^-_\sim$*.*

Proof: Immediate from Proposition 3.1 since with high probability $|S^+| = \Omega(|S|M^+_\sim)$. ∎

Using

$$\mathop{\mathrm{E}}_{\mathbf{x}\in\tilde{\chi}}\{|\mathbf{u}\cdot\mathbf{x}|\} = \mathop{\mathrm{E}}_{\mathbf{x}\in\tilde{\chi}\cap H^+(\mathbf{v})}\{|\mathbf{u}\cdot\mathbf{x}|\}M^+_\sim + \mathop{\mathrm{E}}_{\mathbf{x}\in\tilde{\chi}\cap H^-(\mathbf{v})}\{|\mathbf{u}\cdot\mathbf{x}|\}M^-_\sim$$

we obtain the following:

Corollary 3.3 *With probability at least*

$$1 - e^{-\Omega(\frac{N\min\{M^+_\sim, M^-_\sim\}\gamma^2}{\beta^2})} \ ,$$

$$\begin{aligned}
|(\hat{\mathbf{m}}^+_\sim - \mathbf{m}^+_\sim)\cdot\mathbf{u}| &\le \gamma \mathop{\mathrm{E}}_{\mathbf{x}\in\tilde{\chi}}\{|\mathbf{u}\cdot\mathbf{x}|\} \\
|(\hat{\mathbf{m}}^-_\sim - \mathbf{m}^-_\sim)\cdot\mathbf{u}| &\le \gamma \mathop{\mathrm{E}}_{\mathbf{x}\in\tilde{\chi}}\{|\mathbf{u}\cdot\mathbf{x}|\}
\end{aligned}$$

3.4. Obtaining a Witness

We motivate and sketch our approach. The vector $\mathbf{m}^-(\chi, \mathbf{v})$ has the two desired "witness" properties. It lies in the cone of χ and hence in H ($\mathbf{m}^-(\chi, \mathbf{v}) \cdot \mathbf{h} > 0$) and $\mathbf{m}^-(\chi, \mathbf{v}) \cdot \mathbf{v} < 0$. A desired result would be to show that if $\mathbf{v}$ is bad, these properties also hold for an estimate $\hat{\mathbf{m}}^-(\chi, \mathbf{v})$ achieved using a polynomial number of samples.

Unfortunately, this claim as is not necessarily true. The difficulty is hinted on Section 3.3. A polynomial number of samples can "guarantee" (with high confidence) accuracy of only $\pm \mathrm{E}_\chi(|\mathbf{x} \cdot \mathbf{h}|)/\mathrm{poly}$ for the estimate $|(\hat{\mathbf{m}}^- - \mathbf{m}^-) \cdot \mathbf{h}|$ and similarly, an accuracy of $\pm \mathrm{E}_\chi(|\mathbf{x} \cdot \mathbf{v}|)/\mathrm{poly}$ for $|(\hat{\mathbf{m}}^- - \mathbf{m}^-) \cdot \mathbf{v}|$. On the other hand, there are choices of χ and $\mathbf{v}$ where $\mathbf{m}^- \cdot \mathbf{h}$ (resp., $-\mathbf{m}^- \cdot \mathbf{v}$) are exponentially smaller than $\mathrm{E}\{|\mathbf{x} \cdot \mathbf{h}|\}$ (resp., $\mathrm{E}\{|\mathbf{x} \cdot \mathbf{v}|\}$), and therefore, we can not guarantee that the properties $\hat{\mathbf{m}}^-(\chi, \mathbf{v}) \cdot \mathbf{h} > 0$ and $\hat{\mathbf{m}}^-(\chi, \mathbf{v}) \cdot \mathbf{v} < 0$ hold.

We overcome this problem by restricting the distribution to an appropriate slice of R^n and estimating, instead of $\mathbf{m}^-(\chi, \mathbf{v})$, a linear combination of $\mathbf{m}^-(\chi, \mathbf{v})$ and $\mathbf{m}^+(\chi, \mathbf{v})$. This yields a witness such that with high confidence, its estimate obtained by a polynomial number of samples is also a witness. We elaborate on our approach:

- Observe that $\mathbf{m}^+ \cdot \mathbf{h} + \mathbf{m}^- \cdot \mathbf{h} = \mathrm{E}\{|\mathbf{x} \cdot \mathbf{h}|\}$. Our estimates fail when $\mathbf{m}^- \cdot \mathbf{h}$ is exponentially smaller than $\mathrm{E}\{|\mathbf{x} \cdot \mathbf{h}|\}$. This can be corrected by replacing $\mathbf{m}^-$ by a linear combination of $\mathbf{m}^-$ and $\mathbf{m}^+$. The estimate for such a combination will guarantee membership in H with high probability. We use a vector of the form

$$\mathbf{m}^{-,\nu}(\chi, \mathbf{v}) = (1 - \frac{\nu}{1 - 2\Delta})\mathbf{m}^- + \frac{\nu}{1 - 2\Delta}\mathbf{m}^+ \ ,$$

  which can be estimated using

$$\hat{\mathbf{m}}^{-,\nu}(\chi, \mathbf{v}) = \frac{(1 - \Delta - \nu)\hat{\mathbf{m}}^-_{\tilde{\chi}} + (\Delta + \nu)\hat{\mathbf{m}}^+_{\tilde{\chi}}}{1 - 2\Delta} \ .$$

  Now we have $\mathbf{m}^{-,\nu} \cdot \mathbf{h} \geq \frac{\nu}{1-2\Delta} \mathrm{E}\{|\mathbf{x} \cdot \mathbf{h}|\}$. Therefore, when $\nu = \Omega(1/\mathrm{poly})$, a polynomial number of samples guarantees $\hat{\mathbf{m}}^{-,\nu}(\chi, \mathbf{v}) \cdot \mathbf{h} > 0$.

- However, there is a caveat. The linear combination $\mathbf{m}^{-,\nu}(\chi, \mathbf{v})$ on the original space may be such that $\mathbf{m}^{-,\nu}(\chi, \mathbf{v}) \cdot \mathbf{v} > 0$, and hence, $\hat{\mathbf{m}}^{-,\nu}(\chi, \mathbf{v})$ is unlikely to be a witness. To correct this, we restrict the distributions $\tilde{\chi}$ and χ to a *dense* (of measure at least $1/\mathrm{poly}$) *slice*,

$$\{\mathbf{x} \in S^n | a \leq |\mathbf{x} \cdot \mathbf{v}| \leq a(1 + \mu)\}$$

  (for a properly selected $\mu = \Theta(\epsilon(1 - 2\Delta)/\Delta)$). We select a slice such that $\mathbf{v}$ is bad on the slice (such a slice always exists and can be identified easily). We consider the new distribution χ' obtained by restricting χ to the slice. Note that $a \leq \mathrm{E}_{\mathbf{x} \in \chi'}\{|\mathbf{x} \cdot \mathbf{v}|\} \leq (1 + \mu)a$. Therefore, the dot product $\hat{\mathbf{m}}^{-,\nu}(\chi', \mathbf{v}) \cdot \mathbf{v}$ can be estimated up to additive error of a/poly. Moreover, we claim that for appropriate ν, $-\mathbf{m}^{-,\nu}(\chi', \mathbf{v}) \cdot \mathbf{v} = \Omega(a/\mathrm{poly})$, and hence, with high confidence, $\hat{\mathbf{m}}^{-,\nu}(\chi', \mathbf{v}) \cdot \mathbf{v} < 0$. To see the latter claim, note that

$a(1+\mu) \geq -\mathbf{m}^-(\chi', \mathbf{v}) \cdot \mathbf{v}/M^-(\chi', \mathbf{v}) \geq a$ and $a(1+\mu) \geq \mathbf{m}^+(\chi', \mathbf{v}) \cdot \mathbf{v}/M^+(\chi', \mathbf{v}) \geq a$. Therefore,

$$-\mathbf{m}^{-,\nu}(\chi', \mathbf{v}) \cdot \mathbf{v} > a \left((1 - \frac{\nu}{1-2\Delta}) M^-(\chi', \mathbf{v}) - \frac{\nu}{1-2\Delta}(1+\mu) M^+(\chi', \mathbf{v}) \right) = \Omega(a\epsilon) \ .$$

The latter inequalities follow using the badness of $\mathbf{v}$ on the slice: $M^-(\chi', \mathbf{v}) = \Omega(\epsilon)$ $M^+(\chi', \mathbf{v}) < 1 - \Omega(\epsilon)$ and choosing appropriate $\mu < 1$ and $\nu = \Theta(\epsilon(1-2\Delta))$.

In Section 3.5 we elaborate on choosing a slice and restricting χ to it. In Section 3.6 we consider linear combinations and conclude the proof of Theorem 2.6.

3.5. Choosing a Slice and Restricting χ

Definition 3.4 Consider the vector $\mathbf{v}$ and $0 < a_1 < a_2 < 1$. The (a_1, a_2)-*slice* of the sphere S^n is the set $\{\mathbf{x} \in S^n | a_1 \leq |\mathbf{x} \cdot \mathbf{v}| \leq a_2\}$. The *restricted distribution* χ_{a_1,a_2} is the distribution χ restricted to the (a_1, a_2)-slice. Let $0 \leq M_{a_1,a_2} \leq 1$ be the measure of χ on the (a_1, a_2)-slice. Figure 3 visualizes slices on the 2-dimensional sphere.

Since slices are symmetric and the noise is uniform, $\tilde{\chi}_{a_1,a_2}$ is the same distribution obtained by subjecting χ_{a_1,a_2} to random noise Δ.

To sample from $\tilde{\chi}_{a_1,a_2}$ we sample from $\tilde{\chi}$ repeatedly until we obtain a vector $\mathbf{x}$ that is in the (a_1, a_2)-slice. On average we need $O(M^{-1}_{a_1,a_2})$ samples from $\tilde{\chi}$ to obtain a sample from $\tilde{\chi}_{a_1,a_2}$.

Choosing an appropriate slice. A (a_1, a_2)-slice is *appropriate* when

- For $\mu = \min\{1/4, \frac{\epsilon(1-2\Delta)}{20\Delta}\}$, $a_2 = (1+\mu)a_1$.
- The slice is *fat*, That is, M_{a_1,a_2} is at least $\Omega(\frac{\epsilon(1-2\Delta)}{\text{poly}_1})$, where poly_1 is the negated logarithm of the smallest possible value of $\mathbf{v} \cdot \mathbf{x}$ (for $\mathbf{x} \in \chi$) multiplied by $\log^{-1}(1+\mu)$. We will obtain the vectors $\mathbf{v}$ from the Ellipsoid algorithm. It follows that $M^{-1}_{a_1,a_2}$ is polynomial.
- The slice exhibits the badness behavior of $\mathbf{v}$. That is, if $\mathbf{v}$ is bad, then

$$M^-(\tilde{\chi}_{a_1,a_2}, \mathbf{v}) > (\Delta + \frac{1}{4}\epsilon(1-2\Delta))(M^+(\tilde{\chi}_{a_1,a_2}, \mathbf{v}) + M^-(\tilde{\chi}_{a_1,a_2}, \mathbf{v})) \ . \qquad (13)$$

Proposition 3.5 *If* $\mathbf{v}$ *is bad, then an appropriate slice always exists and can be identified with high confidence in polynomial time.*

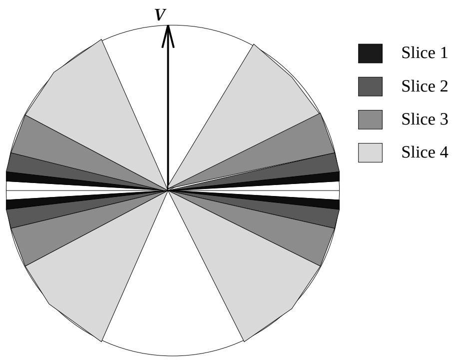

Figure 3: Visualizing slices on the 2-dimensional sphere

Proof: Since all samples are at most b-bits and $\mathbf{v}$ is produced by the Ellipsoid algorithm, the smallest possible (nonzero) magnitude of an inner product is $2^{-\text{poly}'(n,b)}$ for some poly$'$. Consider the $\text{poly}_1 = \log^{-1}(1+\mu)\text{poly}'$ slices defined by

$$a_1 = (1+\mu)^i 2^{-\text{poly}'} \text{ and } a_2 = (1+\mu)^{i+1} 2^{-\text{poly}'} \text{ for } i = 0, \ldots, \lceil \text{poly}_1 \rceil \ .$$

To identify an appropriate slice, we first discard all thin slices. All remaining slices are fat. We select one that exhibits the badness of $\mathbf{v}$. We define *thin* slices as ones with measure at most $\frac{\epsilon(1-2\Delta)}{10\text{poly}_1}$ and *fat* slices as having measure at least $\frac{\epsilon(1-2\Delta)}{5\text{poly}_1}$. We identify and discard all thin slices and retain all fat slices Other slices can go either way. This is performed by sampling r times from $\tilde{\chi}$ while counting the number of times each slice was sampled. We identify as fat and retain only slices that were hit at least $0.15\frac{11\epsilon(1-2\Delta)}{\text{poly}_1}r$ times. A slice is correctly identified if the number of hits is within $\frac{\epsilon(1-2\Delta)}{20\text{poly}_1}$ of the expectation. An application of Chernoff bounds (Proposition 2.1) implies confidence of $1 - 2e^{-2r\left(\epsilon(1-2\Delta)/(20\text{poly}_1)\right)^2}$ for correctly identification of any one slice. Hence, confidence of $1-2\text{poly}_1 e^{-2r\left(\epsilon(1-2\Delta)/(20\text{poly}_1)\right)^2}$ of correctly identifying all slices. Therefore, all slices can be correctly identified with high confidence using a polynomial number of samples.

After discarding all thin slices we identify a slice that exhibits "bad" behaviour with respect to $\mathbf{v}$. Let $M_{\sim}^{-\prime}$ and $M_{\sim}^{+\prime}$ be the measures, on the halfspaces $H^-(\mathbf{v})$ and $H^+(\mathbf{v})$ respectively, of the distribution $\tilde{\chi}$ restricted to the remaining slices. Note that the total measure of χ on discarded slices is at most $\frac{1}{5}\epsilon(1-2\Delta)$. Recall that since v is bad, $M_{\sim}^- \geq \Delta + \frac{1}{2}\epsilon(1-2\Delta)$. Therefore,

$$\frac{M_{\sim}^{-\prime}}{M_{\sim}^{-\prime} + M_{\sim}^{+\prime}} \geq M_{\sim}^- - \frac{1}{5}\epsilon(1-2\Delta) \geq \Delta + \frac{3}{10}\epsilon(1-2\Delta) \ .$$

Recall, from definitions, that

$$M_{\sim}^{-\prime} = \sum_{\text{rem. } (a_1,a_2)\text{-slices}} M_{a_1,a_2} M^-(\tilde{\chi}_{a_1,a_2}, \mathbf{v})$$

$$M_{\sim}^{-\prime} + M_{\sim}^{+\prime} = \sum_{\text{rem. } (a_1,a_2)\text{-slices}} M_{a_1,a_2} \ .$$

Hence, there must exist a remaining slice for which this inequality holds:

$$M^-(\tilde{\chi}_{a_1,a_2}, \mathbf{v}) > \Delta + \frac{3}{10}\epsilon(1-2\Delta) \ .$$

Note that $O(\text{poly})$ samples from $\tilde{\chi}$ are likely to yield samples from each remaining slice. A modification of the arguments in Proposition 2.3 establishes that a slice for which

$$M^-(\tilde{\chi}_{a_1,a_2}, \mathbf{v}) > \Delta + \frac{1}{4}\epsilon(1-2\Delta) \ ,$$

can be identified with high confidence using a polynomial number of samples. ∎

The following proposition is immediate.

Proposition 3.6

$$M^-(\tilde{\chi}_{a_1,a_2}, \mathbf{v})a_1 \leq -\mathbf{m}^-(\tilde{\chi}_{a_1,a_2}, \mathbf{v}) \cdot \mathbf{v} \leq M^-(\tilde{\chi}_{a_1,a_2}, \mathbf{v})a_2$$

$$M^+(\tilde{\chi}_{a_1,a_2}, \mathbf{v})a_1 \leq \mathbf{m}^+(\tilde{\chi}_{a_1,a_2}, \mathbf{v}) \cdot \mathbf{v} \leq M^+(\tilde{\chi}_{a_1,a_2}, \mathbf{v})a_2$$

From here on, when we *restrict* $\tilde{\chi}$ *to a slice* (with respect to $\mathbf{v}$) an appropriate slice is selected as above and outliers are removed. We consider the new distribution χ_{a_1,a_2} of measure 1 on the slice. A sample from $\tilde{\chi}_{a_1,a_2}$ requires an expected number of $M^{-1}_{a_1,a_2}$ samples from $\tilde{\chi}$. That is, there is a polynomial overhead of $M^{-1}_{a_1,a_2}$ in the sampling from $\tilde{\chi}_{a_1,a_2}$.

3.6. Linear Combinations of Centers of Mass

In this subsection we assume that $\tilde{\chi}$ and χ are restricted distribution to an appropriate (a_1, a_2)-slice with respect to $\mathbf{v}$, as outlined in Subsection 3.5.

Consider the following linear combination for $0 \leq \nu < (1-2\Delta)$:

$$\mathbf{m}^{-,\nu}(\chi, \mathbf{v}) = (1 - \frac{\nu}{1-2\Delta})\mathbf{m}^-(\chi, \mathbf{v}) + \frac{\nu}{1-2\Delta}\mathbf{m}^+(\chi, \mathbf{v}) \tag{14}$$

This combination can be viewed as overcompensating by ν for the noise. We choose

$$\nu = \epsilon(1-2\Delta)/20 \ . \tag{15}$$

From Equ. 7– 8 we obtain

$$\mathbf{m}^{-,\nu}(\chi, \mathbf{v}) = \frac{(1-\Delta-\nu)\mathbf{m}^-(\tilde{\chi}, \mathbf{v}) + (\Delta+\nu)\mathbf{m}^+(\tilde{\chi}, \mathbf{v})}{1-2\Delta} \tag{16}$$

We compute the dot product values of the vector $\mathbf{m}^{-,\nu}(\chi, \mathbf{v})$ with $\mathbf{v}$ and with $\mathbf{h}$. It turns out that $\mathbf{m}^{-,\nu}(\chi, \mathbf{v})$ is a witness.

1. From Proposition 3.6 we obtain

$$\begin{aligned}\mathbf{m}^{-,\nu}(\chi,\mathbf{v})\cdot\mathbf{v} &= \frac{(1-\Delta-\nu)\mathbf{m}^{-}(\tilde{\chi},\mathbf{v})\cdot\mathbf{v}+(\Delta+\nu)\mathbf{m}^{+}(\tilde{\chi},\mathbf{v})\cdot\mathbf{v}}{1-2\Delta}\\ &\leq \frac{-(1-\Delta-\nu)M^{-}(\tilde{\chi},\mathbf{v})a_1+(\Delta+\nu)M^{+}(\tilde{\chi},\mathbf{v})a_2}{1-2\Delta}\ .\end{aligned}$$

Using Equ. 13 we obtain

$$\begin{aligned}&\leq \frac{-(1-\Delta-\nu)(\Delta+\frac{1}{4}\epsilon(1-2\Delta))a_1+(\Delta+\nu)(1-\Delta-\frac{1}{4}\epsilon(1-2\Delta))a_2}{1-2\Delta}\\ &\leq a_1\frac{-(1-\Delta-\nu)(\Delta+\frac{1}{4}\epsilon(1-2\Delta))+(\Delta+\nu)(1-\Delta-\frac{1}{4}\epsilon(1-2\Delta))(1+\mu)}{1-2\Delta}\\ &= a_1\left(-\frac{1}{4}\epsilon+\frac{\nu+\mu(\Delta+\nu)(1-\Delta-\frac{1}{4}\epsilon(1-2\Delta))}{1-2\Delta}\right)\\ &\leq a_1\left(-\frac{1}{4}\epsilon+\frac{\nu+\mu(\Delta+\nu)}{1-2\Delta}\right)\ .\end{aligned}$$

Substituting ν (Equ. 15) and $\mu=\min\{1/4,\frac{\epsilon(1-2\Delta)}{20\Delta}\}$ (Section 3.5) we obtain

$$\leq -\frac{\epsilon}{8}a_1\ .$$

2. Using Equ. 14,

$$\begin{aligned}\mathbf{m}^{-,\nu}(\chi,\mathbf{v})\cdot\mathbf{h} &= (1-\frac{\nu}{1-2\Delta})\mathbf{m}^{-}(\chi,\mathbf{v})\cdot\mathbf{h}+\frac{\nu}{1-2\Delta}\mathbf{m}^{+}(\chi,\mathbf{v})\cdot\mathbf{h}\\ &\geq \min\left\{1-\frac{\nu}{1-2\Delta},\frac{\nu}{1-2\Delta}\right\}(\mathbf{m}^{-}(\chi,\mathbf{v})\cdot\mathbf{h}+\mathbf{m}^{+}(\chi,\mathbf{v})\cdot\mathbf{h})\\ &= \min\left\{1-\frac{\nu}{1-2\Delta},\frac{\nu}{1-2\Delta}\right\}\operatorname*{E}_{\mathbf{x}\in\chi}\{\mathbf{x}\cdot\mathbf{h}\}\\ &\geq \frac{\epsilon}{10}\operatorname*{E}_{\mathbf{x}\in\chi}\{\mathbf{x}\cdot\mathbf{h}\}\ .\end{aligned}$$

Consider the estimate:

$$\hat{\mathbf{m}}^{-,\nu}(\chi,\mathbf{v})=\frac{(1-\Delta-\nu)\hat{\mathbf{m}}^{-}(\tilde{\chi},\mathbf{v})+(\Delta+\nu)\hat{\mathbf{m}}^{+}(\tilde{\chi},\mathbf{v})}{1-2\Delta} \tag{17}$$

We first bound the estimation accuracy of the dot products of $\hat{\mathbf{m}}^{+}_{\sim}=\hat{\mathbf{m}}^{+}(\tilde{\chi},\mathbf{v})$ and $\hat{\mathbf{m}}^{-}_{\sim}=\hat{\mathbf{m}}^{+}(\tilde{\chi},\mathbf{v})$ with $\mathbf{v}$ and $\mathbf{h}$ (using Section 3.3). From there we deduce the estimation accuracy for the dot products $\hat{\mathbf{m}}^{-,\nu}(\chi,\mathbf{v})\cdot\mathbf{v}$ and $\hat{\mathbf{m}}^{-,\nu}(\chi,\mathbf{v})\cdot\mathbf{h}$ and establish that with a polynomial amount of sampling, the estimation errors can be made small with respect to the dot product values. Hence, with high confidence, $\hat{\mathbf{m}}^{-,\nu}(\chi,\mathbf{v})$ is a witness.

Proposition 3.7 *If the estimates $\hat{\mathbf{m}}^{-}(\tilde{\chi},\mathbf{v})$ and $\hat{\mathbf{m}}^{+}(\tilde{\chi},\mathbf{v})$ are obtained using N samples then with confidence at least*

$$1-e^{-\Omega(\frac{N(1-2\Delta)^2\gamma^2}{\beta^2})}\ ,$$

$$|\hat{\mathbf{m}}^{-,\nu}(\chi,\mathbf{v})\cdot\mathbf{u}-\mathbf{m}^{-,\nu}(\chi,\mathbf{v})\cdot\mathbf{u}|\leq\gamma\operatorname*{E}_{\mathbf{x}\in\chi}\{|\mathbf{x}\cdot\mathbf{u}|\}\ .$$

Proof: Immediate from Corollary 3.3 and Equ. 17. We assume without loss of generality that $\min\{M^+_{\sim}, M^-_{\sim}\} \geq \Delta \geq 1/4$. (It is easy to introduce additional random noise and increase the noise level to $\max\{\Delta, 1/4\}$). ∎

To utilize Proposition 3.7, we bound $\mathrm{E}_{\tilde{\chi}}\{|\mathbf{u}\cdot\mathbf{x}|\}$ for $\mathbf{u} = \mathbf{v}$ and for $\mathbf{u} = \mathbf{h}$. Note that $\mathrm{E}_{\mathbf{x}\in\tilde{\chi}}\{|\mathbf{u}\cdot\mathbf{x}|\} = \mathrm{E}_{\mathbf{x}\in\chi}\{|\mathbf{u}\cdot\mathbf{x}|\}$.

If $\mathbf{u} = \mathbf{v}$ then, using Proposition 3.6,

$$\mathop{\mathrm{E}}_{\mathbf{x}\in\chi}\{|\mathbf{v}\cdot\mathbf{x}|\} = \mathbf{m}^+(\tilde{\chi},\mathbf{v})\cdot\mathbf{v} - \mathbf{m}^-(\tilde{\chi},\mathbf{v})\cdot\mathbf{v} \leq a_2 \leq 1.5a_1 \; . \tag{18}$$

If $\mathbf{u} = \mathbf{h}$ then

$$\mathop{\mathrm{E}}_{\mathbf{x}\in\chi}\{|\mathbf{h}\cdot\mathbf{x}|\} = \mathop{\mathrm{E}}_{\mathbf{x}\in\chi}\{\mathbf{h}\cdot\mathbf{x}\} \; . \tag{19}$$

We need to choose N large enough such that with high confidence the estimation error is small with respect to the magnitudes, that is,

$$\begin{aligned} |\hat{\mathbf{m}}^{-,\nu}(\chi,\mathbf{v})\cdot\mathbf{v} - \mathbf{m}^{-,\nu}(\chi,\mathbf{v})\cdot\mathbf{v}| &< \frac{\epsilon a_1}{8} & (20)\\ |\hat{\mathbf{m}}^{-,\nu}(\chi,\mathbf{v})\cdot\mathbf{h} - \mathbf{m}^{-,\nu}(\chi,\mathbf{v})\cdot\mathbf{h}| &< \frac{\epsilon}{10}\mathop{\mathrm{E}}_{\mathbf{x}\in\chi}\{\mathbf{x}\cdot\mathbf{h}\} \; . & (21)\end{aligned}$$

From Propositions 3.7 and Equ. 18 and 19 we obtain that for any $\gamma > 0$, with confidence at least

$$1 - e^{-\Omega(\frac{N(1-2\Delta)^2\gamma^2}{\beta^2})} \; ,$$

we have

$$\begin{aligned} |\hat{\mathbf{m}}^{-,\nu}(\chi,\mathbf{v})\cdot\mathbf{v} - \mathbf{m}^{-,\nu}(\chi,\mathbf{v})\cdot\mathbf{v}| &< 1.5\gamma a_1 & (22)\\ |\hat{\mathbf{m}}^{-,\nu}(\chi,\mathbf{v})\cdot\mathbf{h} - \mathbf{m}^{-,\nu}(\chi,\mathbf{v})\cdot\mathbf{h}| &< \gamma\mathop{\mathrm{E}}_{\mathbf{x}\in\chi}\{\mathbf{x}\cdot\mathbf{h}\} & (23)\end{aligned}$$

From Ineq. 20 and 22 we get $\gamma \leq \frac{\epsilon}{12}$ and from Ineq. 21 and 23 we obtain $\gamma \leq \frac{\epsilon}{10}$. Therefore, if $\gamma = \epsilon/13$ then for N samples, $\hat{\mathbf{m}}^{-,\nu}(\chi,\mathbf{v})$ is a witness with confidence

$$1 - e^{-\Omega(\frac{N(1-2\Delta)^2\epsilon^2}{\beta^2})} \; .$$

Hence, a polynomial size N guarantees very high confidence. This concludes the proof of Theorem 2.6.

3.7. Incorporating an unknown noise level

Standard techniques [11] can be used to apply our algorithm to the case where the noise level Δ is not given. We outline an alternative cleaner approach that incorporates unknown noise level into our algorithm. The basic idea (also mentioned in [4]) is to overcompensate for the noise when selecting a witness. The algorithm works in phases and maintains an interval $[\Delta_{\min}, \Delta_{\max}]$ for the possible "true" Δ. The running time of each phase is polynomial in

$1/(1-2\Delta_{\max})$. If $\Delta \in [\Delta_{\min}, \Delta_{\max}]$, the algorithm finds (with high probability) a good classifier in the current phase. Otherwise, if $\Delta > \Delta_{\max}$, the algorithm progresses to the next phase with the interval $[\Delta_{\max}, \Delta_{\max}/2 + 1/4]$ for the noise, and henceforth. (Each phase halves the distance between $\Delta_{\max}$ and $1/2$.) We elaborate on the execution of each phase.

- **Witness identification:** We use $\Delta_{\max}$ as a guide to determine the amount of sampling, that is, we utilize enough samples for the algorithm to be correct with high probability for any $\Delta \leq \Delta_{\max}$. When choosing a slice, we take the *appropriate* slice that is fat with $M_{a_1,a_2} = \Omega(\frac{\epsilon(1-2\Delta_{\max})}{\text{poly}_1})$ and maximizes the (estimated) ratio $M^-_{\sim}/M^+_{\sim}$. Consequently, to select a witness we use $\hat{\mathbf{w}} = (1-\alpha)\hat{\mathbf{m}}^-_{\sim} + \alpha\hat{\mathbf{m}}^+_{\sim}$ and we choose α to be such that $\hat{\mathbf{w}} \cdot \mathbf{v} = 0$. Intuitively, we "push" the second witness property as far as we can without violating it, and this can only "help" the first witness property. These choices will assure with high confidence that $\hat{\mathbf{w}} \cdot \mathbf{h} > 0$ if indeed $\Delta \leq \Delta_{\max}$ and initially, $\mathbf{v}$ was bad.

- **Ellipsoid algorithm:** The main difficulty is that not knowing Δ, we can not distinguish between larger value of Δ or the current vector $\mathbf{v}$ being bad. We record Δ' as the highest noise level for which the current vector $\mathbf{v}$ is good. We can stop if $\Delta' \leq \Delta_{\min}$ or when the volume of the remaining Ellipsoid is smaller than our bound on the feasible region. Otherwise, we assume that $\mathbf{v}$ is bad and continue running the Ellipsoid algorithm. If the remaining volume is below our bound, it implies that witness identification failed. This could be contributed to either (i) insufficient sampling (since $\Delta > \Delta_{\max}$) or (ii) that one of the vectors assumed bad in a previous iteration was actually good (since $\Delta \leq \Delta'$). To distinguish between these too reasons, we consider the classifier $\mathbf{v}''$ that minimized the respective Δ' over all previous iterations (did best so far). We denote by Δ'' the corresponding minimum Δ'. If $\Delta'' \leq \Delta_{\max}$ we stop and output $\mathbf{v}''$ as a good classifier. Otherwise, we deduce that $\Delta > \Delta_{\max}$ and proceed to the next phase.

3.8. Scaling instead of slicing

We outline an alternative algorithm to find a witness for a bad vector $\mathbf{v}$. This approach was instrumental in extending our results to data with some non-uniform noise distributions.

1. We transform the distribution by scaling vectors such that all vectors have a dot product value of $\{-1, 1\}$ with $\mathbf{v}$. We apply the transformation

$$T(\mathbf{x}) = \frac{\mathbf{x}}{||\mathbf{x}|||\mathbf{x} \cdot \mathbf{v}|} .$$

   Note that $T(\tilde{\chi})$ is the same distribution obtained by applying Δ noise to $T(\chi)$.

2. We apply "outliers removal" to the scaled distribution $T(\tilde{\chi})$.

3. Consider the vectors $\mathbf{m}_{\sim}^{-} \equiv \mathbf{m}^{-}(T(\tilde{\chi}), \mathbf{v})$, $\mathbf{m}_{\sim}^{+} \equiv \mathbf{m}^{+}(T(\tilde{\chi}), \mathbf{v})$, $\mathbf{m}^{-} \equiv \mathbf{m}^{-}(T(\chi), \mathbf{v})$, and $\mathbf{m}^{+} \equiv \mathbf{m}^{+}(T(\chi), \mathbf{v})$. Our candidate witness is a vector of the form

$$\mathbf{w} = \frac{(1-\Delta-\nu(1-2\Delta))\mathbf{m}_{\sim}^{-} + (\Delta+\nu(1-2\Delta))\mathbf{m}_{\sim}^{+}}{1-2\Delta} = (1-\nu)\mathbf{m}^{-} + \nu\mathbf{m}^{+}$$

(for $\nu > 0$ to be determined as follows). We have

$$\mathbf{w} \cdot \mathbf{h} \geq \nu(\mathbf{m}^{-} + \mathbf{m}^{+}) \cdot \mathbf{h} = \nu \mathop{\mathrm{E}}_{\mathbf{x} \in T(\chi)} \{\mathbf{x} \cdot \mathbf{h}\} = \nu \mathop{\mathrm{E}}_{\mathbf{x} \in T(\tilde{\chi})} \{|\mathbf{x} \cdot \mathbf{h}|\} \ .$$

and

$$\mathbf{w} \cdot \mathbf{v} = (1-\nu)M^{-} + \nu M^{+} \leq -(1-\nu)\epsilon/2 + \nu \ .$$

Hence, for $\nu \leq \frac{\epsilon}{2+\epsilon}$, $\mathbf{w}$ is a witness. For $\nu \geq 1/\text{poly}$, an estimate obtained with sufficiently but polynomially many samples will be such that $\hat{\mathbf{w}} \cdot \mathbf{h} > 0$, with very high confidence (see estimation accuracy discussion in Subsection 3.3). Using the fact that $\mathrm{E}_{\mathbf{x} \in T(\chi)}\{|\mathbf{x} \cdot \mathbf{v}|\} = 1$, we obtain that for $\nu < \epsilon/3$,

$$|\mathbf{w} \cdot \mathbf{v}| \geq \Theta(\epsilon) = \Theta(\epsilon) \mathop{\mathrm{E}}_{\mathbf{x} \in T(\chi)} \{|\mathbf{x} \cdot \mathbf{v}|\} \ .$$

Hence, for $1/\text{poly} \leq \nu \leq \epsilon/3$, the estimate $\hat{\mathbf{w}}$ (obtained with polynomially many samples) is a witness with very high confidence. (note, however, that since we can explicitly test $\hat{\mathbf{w}} \cdot \mathbf{v}$, high confidence for $\hat{\mathbf{w}} \cdot \mathbf{v} < 0$ is not necessary.)

An unknown noise level Δ can be incorporated into the algorithm in a similar fashion as discussed in Subsection 3.7. To obtain a witness we first compute estimates $\hat{\mathbf{m}}_{\sim}^{-}$ and $\hat{\mathbf{m}}_{\sim}^{+}$ (using enough samples for $\Delta \leq \Delta_{\max}$), choose $\alpha = |\hat{\mathbf{m}}_{\sim}^{-} \cdot \mathbf{v} / \hat{\mathbf{m}}_{\sim}^{+} \cdot \mathbf{v}|$, and use $\hat{\mathbf{w}} = \hat{\mathbf{m}}_{\sim}^{-} + \alpha \hat{\mathbf{m}}_{\sim}^{+}$ as a witness. The claims above establish that $\hat{\mathbf{w}} \cdot \mathbf{h} > 0$ with high probability (if indeed $\Delta \leq \Delta_{\max}$ and sufficient sampling was performed).

4. Employing the Ellipsoid Algorithm

Let H and F be as in Subsection 2.3. Consider the linear system of inequalities:

$$\text{Find } \mathbf{g} \in R^n, \text{ subject to: } \quad \forall \mathbf{x} \in H, \ \mathbf{g} \cdot \mathbf{x} > 0 \ . \tag{24}$$

The set F is the feasible region of this system. Note that the number of inequalities is very large and none are explicitly provided, since H is not known.

The Ellipsoid algorithm [12] can solve such a system, given a *separation oracle* (see, e.g. [8, 17] for background). When the separation oracle is presented with a point $\mathbf{v} \in R^n$, it either determines that $\mathbf{v}$ is feasible and terminates, or returns a violated constraint. In our case, a violated constraint is a vector $\mathbf{w} \in H$ such that $\mathbf{w} \cdot \mathbf{v} < 0$.

The Ellipsoid algorithm is defined for non-strict inequalities, but can be easily modified to handle strict inequalities as is the case here (strict inequalities mean that we need to find an interior point in the feasible region of the equivalent system with non-strict inequalities). In our problem, we need to avoid the trivial vector $\mathbf{v} = \mathbf{0}$. (It is the only vector that does not yield a violated constraint and is not a good vector.) The trivial vector can be excluded by perturbing slightly when encountered or by artificially adding the constraint $w_i > 1/n$ or $w_i < -1/n$ for some $1 \leq i \leq n$. (if $F \neq \{\mathbf{0}\}$, then there must be a solution for at least one choice of i and one of the two constraints).

The feasible set F of the system (24) contains vectors that separate χ with *no error* (F contains good vectors with $\epsilon = 0$). Our objective is the more moderate goal of finding a good vector for inverse-polynomially large ϵ (see Definition 2.2). The feasible set F is contained in the set of good vectors. We use the following separation oracle.

Modified Separation Oracle. Let $\mathbf{v}$ be the center of the current Ellipsoid.

- Test if $\mathbf{v}$ is a good vector. Proposition 2.3 establishes that this can be performed with high confidence in polynomial time. If $\mathbf{v}$ is a good vector, stop the Ellipsoid algorithm and return $\mathbf{v}$. Note that $\mathbf{v}$ is not necessarily in F.
- Otherwise, $\mathbf{v}$ is bad. Apply Theorem 2.6 to find, with high confidence and in polynomial time, a witness $\mathbf{w}$ for the badness of $\mathbf{v}$. By definition, the witness is $\mathbf{w} \in H$ such that $\mathbf{v} \cdot \mathbf{w} < 0$. This is a violated inequality of the system (24) and it allows to continue executing the Ellipsoid Algorithm.

The algorithm is applied in conjunction with the modified separation oracle.

The Ellipsoid algorithm initializes by bounding a portion or all of feasibility region by an Ellipsoid E_0. In our problem, the feasibility region is initially bounded by the unit ball B^n. Therefore, an initial Ellipsoid of volume $O((2e\pi/n)^{n/2}/\sqrt{\pi n})$ is used. (All we need is for the volume to be $\leq 2^{\text{poly}}$)

Correctness:. Either a good vector is found during the execution of the Ellipsoid algorithm, or else, it is carried to termination and finds a vector in F, which is a good vector for any ϵ. Each application of the separation oracle is correct with high confidence, therefore, if there are polynomially many iterations, all of them are correct with high confidence. (For correctness we can tolerate even e^{poly} many iterations).

Polynomial time:. To establish a polynomial running time it suffices to show that a good vector is found within polynomially many iterations.

In each iteration, the Ellipsoid algorithm applies the separation oracle to the center $\mathbf{v}$ of the current Ellipsoid E_i. The violated constraint returned by the oracle is used to obtain a new Ellipsoid E_{i+1} that also encloses the feasibility region, but its volume $\text{vol}(E_{i+1})$ is smaller by at least $\Omega(1/n)$ fraction than the volume $\text{vol}(E_i)$, that is, $\frac{\text{vol}(E_{i+1})}{\text{vol}(E_i)} \leq 1 - \Omega(1/n)$.

The Ellipsoid algorithm terminates in polynomially many iteration when the volume of the feasibility polyhedron is at least exponent of a negated polynomial in the volume of E_0. This is established in Proposition 2.4.

5. Non-Uniform Noise Distributions

We analyse the behaviour of variants of our algorithm on data where the noise level is non-uniform and not given to us. It turns out that our results extend to realistic non-uniform noise distributions, where the noise level is correlated with proximity to optimal threshold function.

Let $\Delta(\mathbf{x})$ be the *noise function.* The density function of the noisy distribution of examples is

$$\tilde{\chi}(\mathbf{x}) = (1 - \Delta(\mathbf{x}))\chi(\mathbf{x}) + \Delta(-\mathbf{x})\chi(-\mathbf{x}) \ .$$

The respective centers of mass of $\tilde{\chi}$ on the halfspaces determined by $\mathbf{v}$ are denoted

$$\begin{aligned} \mathbf{m}_{\sim}^{-} &= \int_{H^-} \mathbf{x}\tilde{\chi}(\mathbf{x})d\mathbf{x} \\ \mathbf{m}_{\sim}^{+} &= \int_{H^+} \mathbf{x}\tilde{\chi}(\mathbf{x})d\mathbf{x} \end{aligned}$$

The first issue is to distinguish good and bad vectors in the presence of nonuniform noise. We assume that there is an upper bound on the noise $\forall \mathbf{x} \in \chi, \Delta(\mathbf{x}) \le \Delta_{\max} < 1/2$. To guarantee error at most ϵ in the classification performed according to a good vector, we apply the criteria as in Subsection 2.1 where $\Delta_{\max}$ is substituted for Δ. The amount of sampling needed is polynomial in $1/(1 - 2\Delta_{\max})$. Note that the assumption that $\Delta_{\max}$ is provided can be bypassed with the methods discussed earlier on.

The remaining (and challenging) issue is identifying witnesses for bad vectors in the presence of nonuniform noise. Consider a vector of the form

$$\mathbf{z}_\alpha = (1 - \alpha)\mathbf{m}_{\sim}^{-} + \alpha\mathbf{m}_{\sim}^{+} \ .$$

By substituting we get

$$\mathbf{z}_\alpha = (1-\alpha)\int_{H^-} \mathbf{x}\chi(\mathbf{x})d\mathbf{x} - \int_{H^-} \mathbf{x}\Delta(\mathbf{x})\chi(\mathbf{x})d\mathbf{x} + \alpha\int_{H^+} \mathbf{x}\chi(\mathbf{x})d\mathbf{x} - \int_{H^+} \mathbf{x}\Delta(\mathbf{x})\chi(\mathbf{x})d\mathbf{x} \ .$$

A sufficient condition for $\mathbf{z}_\alpha \cdot \mathbf{h} > 0$ is

$$\alpha \ge \frac{\mathrm{E}_{\mathbf{x}\in\chi\cap H^+}\{\mathbf{x}\cdot\mathbf{h}\Delta(\mathbf{x})\}}{\mathrm{E}_{\mathbf{x}\in\chi\cap H^+}\{\mathbf{x}\cdot\mathbf{h}\}} \ .$$

We consider now conditions for $\mathbf{z}_\alpha \cdot \mathbf{v} < 0$.

$$\mathbf{z}_\alpha \cdot \mathbf{v} = \int_{H^-} \mathbf{x}\cdot\mathbf{v}(1 - \alpha - \Delta(\mathbf{x}))\chi(\mathbf{x})d\mathbf{x} + \int_{H^+} \mathbf{x}\cdot\mathbf{v}(\alpha - \Delta(\mathbf{x}))\chi(\mathbf{x})d\mathbf{x} \ .$$

Note that the first summand is negative and the second positive. A sufficient condition for $\mathbf{z}_\alpha \cdot \mathbf{v} < 0$ is

$$\alpha \le \frac{\mathrm{E}_{\mathbf{x}\in\chi\cap H^+}\{\mathbf{x}\cdot\mathbf{v}\Delta(\mathbf{x})\}}{\mathrm{E}_{\mathbf{x}\in\chi\cap H^+}\{\mathbf{x}\cdot\mathbf{v}\}} \ .$$

Suppose α is such that $\mathbf{z}_\alpha \cdot \mathbf{v} = 0$ (that is, maximum for which $\mathbf{z}_\alpha \cdot \mathbf{v} \le 0$). A sufficient condition for $\mathbf{z}_\alpha$ to be a witness is the following:

For every bad vector $\mathbf{v}$ *and a perfect classifier* $\mathbf{h}$ *we have*

$$\frac{\mathrm{E}_{\mathbf{x}\in\chi\cap H^+}\{\mathbf{x}\cdot\mathbf{h}\Delta(\mathbf{x})\}}{\mathrm{E}_{\mathbf{x}\in\chi\cap H^+}\{\mathbf{x}\cdot\mathbf{h}\}} \le \frac{\mathrm{E}_{\mathbf{x}\in\chi\cap H^+}\{\mathbf{x}\cdot\mathbf{v}\Delta(\mathbf{x})\}}{\mathrm{E}_{\mathbf{x}\in\chi\cap H^+}\{\mathbf{x}\cdot\mathbf{v}\}} . \tag{25}$$

An interpretation of the latter condition is that the noise level is more highly correlated with proximity to the threshold function specified by $\mathbf{h}$ (the hyperplane $\mathbf{h}\cdot\mathbf{x}=0$) than with proximity to the threshold of $\mathbf{v}$. This analysis also indicates the limitations of the approach of using such a $\mathbf{z}_\alpha$ as a witness. For some noise functions $\Delta(\mathbf{x})$, the vector $\mathbf{z}_\alpha$ is not a witness for *any* value of α.

5.1. Good estimation of a witness

As is the case for uniform-noise, however, even if $\mathbf{z}_\alpha$ constitutes a witness, there are scenarios where an estimate $\hat{\mathbf{z}}_\alpha$ obtained with polynomially many samples, may not constitute a witness with high probability. A condition for *good estimation* of $\mathbf{z}_\alpha\cdot\mathbf{h}$ (that is, the estimate has positive dot product with $\mathbf{h}$ with high probability) is that for some polynomial poly$'$,

$$\mathbf{z}_\alpha\cdot\mathbf{h} \ge \mathop{\mathrm{E}}_{\mathbf{x}\in\chi}\{\mathbf{x}\cdot\mathbf{h}\}/\mathrm{poly}' .$$

Equivalently, when for some poly_1 and poly_2

$$\int_{\mathbf{x}\in\chi||\alpha-\Delta(\mathbf{x})|>1/\mathrm{poly}_1} \mathbf{x}\cdot\mathbf{h}\chi(\mathbf{x})d\mathbf{x} \ge \mathop{\mathrm{E}}_{\mathbf{x}\in\chi}\{\mathbf{x}\cdot\mathbf{h}\}/\mathrm{poly}_2 .$$

(Intuitively, the bad scenario for "good estimation" is when the samples that "cancel out" in the linear combination $\mathbf{z}_\alpha$ are the ones that dominate the expected value of $\mathbf{x}\cdot\mathbf{h}$.) Let α_0 be such that $\mathbf{z}_{\alpha_0}\cdot\mathbf{v}=0$. It follows that a sufficient condition for "good estimation" of $\mathbf{z}_\alpha\cdot\mathbf{h}$ is that there exists a polynomial poly$''$ such that

$$\alpha_0 \ge \frac{\mathrm{E}_{\mathbf{x}\in\chi\cap H^+}\{\mathbf{x}\cdot\mathbf{h}\Delta(\mathbf{x})\}}{\mathrm{E}_{\mathbf{x}\in\chi\cap H^+}\{\mathbf{x}\cdot\mathbf{h}\}} + 1/\mathrm{poly}'' . \tag{26}$$

We sketch how to select an appropriate value α. For any polynomial poly, we can select α which is "good" with probability $(1-1/\mathrm{poly})$ by choosing uniformly at random an integer $i<\mathrm{poly}$, using $\alpha=\hat{\alpha}_0+i/(\mathrm{poly}\cdot\mathrm{poly}'')$, and increasing the amount of sampling accordingly.

As was the case for uniform noise, to achieve good estimation (Equ. 26), we can apply *slicing* or *scaling* of the distribution. The difficulty in using slices is in identifying a slice such that the vector $\mathbf{v}$ is bad with respect to the distribution restricted to a slice. Since the noise distribution is not provided, we do not know how to distinguish between slices with higher average level of noise and slices on which $\mathbf{v}$ is bad. Note that $\mathbf{v}$ must be bad on the selected slice for Equ. 26 to hold. Scaling, fortunately, is applicable. Below we establish a sufficient condition for noise distributions (similar to Equ. 25) where a scaling approach yields a witness with good estimation.

5.2. Use of scaling with non-uniform noise

Suppose the distribution is *scaled* (and outliers are subsequently removed as in Subsection 3.8). Let χ' be the scaled χ. We have $\forall \mathbf{x} \in \chi', |\mathbf{x} \cdot \mathbf{v}| = 1$. We get

$$\mathbf{z}_\alpha \cdot \mathbf{v} = -\int_{H^-} (1 - \alpha - \Delta(\mathbf{x}))\chi'(\mathbf{x})d\mathbf{x} + \int_{H^+} (\alpha - \Delta(\mathbf{x}))\chi'(\mathbf{x})d\mathbf{x} \ .$$

Manipulations (and using $\alpha \leq 1/2$) yield

$$\mathbf{z}_\alpha \cdot \mathbf{v} \leq -(\frac{1}{2} - \Delta_{\max})M^- + \left(\alpha - \mathop{\mathrm{E}}_{\mathbf{x}\in\chi'\cap H^+}\{\Delta(\mathbf{x})\}\right) M^+ \ .$$

Hence, the value of α for which $\mathbf{z}_\alpha \cdot \mathbf{v} = 0$ is

$$\alpha_0 \geq \mathop{\mathrm{E}}_{\mathbf{x}\in\chi'\cap H^+}\{\Delta(\mathbf{x})\} + \frac{(\frac{1}{2} - \Delta_{\max})M^-}{M^+} \ .$$

Note that badness of $\mathbf{v}$ implies that the last summand is at least 1/poly. Hence, when scaling is used in conjunction with the witness selection condition above, we obtain a sufficient condition for $\mathbf{z}_{\alpha_0}$ to be a witness and with high probability for the estimate $\hat{\mathbf{z}}_{\alpha_0}$ to be a witness.

For every bad vector $\mathbf{v}$ *and a perfect classifier* $\mathbf{h}$*,*

$$\frac{\mathrm{E}_{\mathbf{x}\in\chi\cap H^+}\{\frac{\mathbf{x}\cdot\mathbf{h}}{\mathbf{x}\cdot\mathbf{v}}\Delta(\mathbf{x})\}}{\mathrm{E}_{\mathbf{x}\in\chi\cap H^+}\{\frac{\mathbf{x}\cdot\mathbf{h}}{\mathbf{x}\cdot\mathbf{v}}\}} \leq \mathop{\mathrm{E}}_{\mathbf{x}\in\chi'\cap H^+}\{\Delta(\mathbf{x})\} + 1/\mathrm{poly}_1 = \mathop{\mathrm{E}}_{\mathbf{x}\in\chi\cap H^+}\{\Delta(\mathbf{x})\} + 1/\mathrm{poly}_1 \ . \tag{27}$$

where $\mathrm{poly}_1 > 2\frac{M^+}{(\frac{1}{2}-\Delta_{\max})M^-}$ *is some polynomial.*

This means that for any pair of a perfect threshold $\mathbf{h}$ and a bad vector $\mathbf{v}$, the noise level is more highly correlated with proximity to the threshold function specified by $\mathbf{h}$ (the hyperplane $\mathbf{h} \cdot \mathbf{x} = 0$) than with proximity to the hyperplane $\mathbf{v} \cdot \mathbf{x} = 0$. The above condition is a realistic assumption on the behaviour of noise.

5.3. Probabilistically consistent noise

We discuss how the conditions in Equations 25 and 27 relate to the noise-distribution model suggested and investigated by Bylander [6]. Bylander named his model *probabilistically consistent*, which refers to noise which is non-decreasing with proximity to one optimal threshold $\mathbf{h}$. In particular, the noise is radially symmetric. It seems that neither class of noise functions generalizes the other. The conditions on the noise in Equ. 27 do not imply symmetry or monotonicity of noise-level with proximity to threshold. On the other hand, there are simple examples of probabilistically consistent data for which Equ. 27 does not hold. For example, consider a distribution χ with 3 equally probable points $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3$, $\mathbf{h} = (1, 0)$, and $\mathbf{v} = (0.996, -0.09)$ as follows:

$\mathbf{x}$	$\Delta(\mathbf{x})$	$\mathbf{x}\cdot\mathbf{h}$	$\mathbf{x}\cdot\mathbf{v}$
$\mathbf{x}_1 = (0.1, 0.995)$	0.25	0.1	0.010
$\mathbf{x}_2 = (0.4, 0.917)$	0	0.4	0.316
$\mathbf{x}_3 = (0.05, 0.999)$	0.3	0.05	−0.040

Note that $\{\mathbf{x}_1, \mathbf{x}_2\} \subset H^+$ and $\mathbf{x}_3 \in H^-$. It is easy to verify that this data is probabilistically consistent, but Equ. 25 and 27 do not hold. We remark that the other requirement that the noise is nondecreasing with respect to a *specified single optimal threshold* $\mathbf{h}$ (in contrast, we are interested in Equ. 27 to hold for several optimal vectors) is only a technicality.

5.4. Use of slices with non-uniform noise

We remark that if we were able to determine if a vector $\mathbf{v}$ is bad with respect to χ restricted to a slice, then we could obtain a polynomial time algorithm that is applicable to a generalization of "probabilistically consistent" noise functions. We observe that if the distribution is restricted to a slice (see Subsection 3.5), the conditions in Equ. 25 and 27 can be relaxed to

$$\frac{\mathrm{E}_{\mathbf{x}\in\chi\cap H^+}\{\mathbf{x}\cdot\mathbf{h}\Delta(\mathbf{x})\}}{\mathrm{E}_{\mathbf{x}\in\chi\cap H^+}\{\mathbf{x}\cdot\mathbf{h}\}} \leq \mathop{\mathrm{E}}_{\mathbf{x}\in\chi\cap H^+}\{\Delta(\mathbf{x})\}(1+\mu)^{-1}(\ +\frac{1}{(1+\mu)\mathrm{poly}_1} \text{ for Equ. 27}) \ . \tag{28}$$

It is easy to see that for any "probabilistically consistent" noise functions (with respect to a perfect classifier $\mathbf{h}$) we have:

$$\frac{\mathrm{E}_{\mathbf{x}\in\chi\cap H^+}\{\mathbf{x}\cdot\mathbf{h}\Delta(\mathbf{x})\}}{\mathrm{E}_{\mathbf{x}\in\chi\cap H^+}\{\mathbf{x}\cdot\mathbf{h}\}} \leq \mathop{\mathrm{E}}_{\mathbf{x}\in\chi\cap H^+}\{\Delta(\mathbf{x})\} \ .$$

Simple arguments show that the relaxed Equ. 27 holds for a suitable choice of μ (we need $1/\mu <$ poly, for some polynomial poly) and with respect to some applicable choices of $\mathbf{h}_1, \ldots, \mathbf{h}_k$ (see Subsection 2.3).

6. Summary and relation to previous work

After laying out our algorithm, we provide some review of previous algorithms for the problem, in particular the perceptron algorithm and noise-tolerant variants, and point on relation between the ideas and difficulties encountered there and our algorithm.

The perceptron algorithm [16, 1] is a simple iterative algorithm that maintains a "current classifier" $\mathbf{g}$ and views a stream of examples. When it encounters an example $\mathbf{x}$ which is misclassified by $\mathbf{g}$, the algorithm replaces the current classifier $\mathbf{g}$ by a linear combination of $\mathbf{g}$ and $\mathbf{x}$. The perceptron algorithm converges for noise-free *linearly separable data* (a linear threshold is guaranteed to exist). The convergence rate depends on the *separation parameter* which is the maximum σ such that there exist a vector $\mathbf{h}$ such that

$$\forall \mathbf{x} \in \chi,\ \frac{\mathbf{h}\cdot\mathbf{x}}{||\mathbf{h}||||\mathbf{x}||} > \sigma \ .$$

The number of iterations performed by the perceptron algorithm is polynomial in $1/\sigma$ [14]. The separation parameter σ, however, can be exponentially small. Hence, the perceptron algorithm is not a polynomial time algorithm.

The convergence proof of the perceptron algorithm assumes correct labeling of examples, and hence, is not directly applicable to noisy data. Bylander [5] considered data subject to random classification noise and provided a modified perceptron algorithm. He showed that with the additional requirement that the data is well separated with respect to *any* hyperplane, that is,

$$\forall \mathbf{u} \mathop{\mathrm{E}}_{\mathbf{x}\in\mathcal{X}} \{|\mathbf{u}\cdot\mathbf{x}|\} \geq \sigma||\mathbf{u}||,$$

his algorithm converges using number of iteration polynomial is $1/\sigma$. In our terminology, Bylander's algorithm constructs, in each iteration, a *witness* for the current classifier. (Here the witness must have cosine at most $-\sigma$ with the current classifier.) It then applies the original perceptron algorithm using the witness as a misclassified example. Bylander algorithm constructs a witness using a linear combination of the two averages of examples with negative cosines and positive cosines with $\mathbf{u}$ (which is reminiscent of what we are doing).

Blum *et al.* [4] showed that when data is subject to random classification noise, perceptrons can be learned in polynomial time by relaxing the hypothesis not to be a single linear threshold but a decision rule consisting of linear thresholds. Their algorithm is iterative and mimics Bylander's modified perceptron algorithm in each iteration. Essentially, in each iteration, they find a classifier which is good on all examples $\mathbf{x}$ where $|\mathbf{u}\cdot\mathbf{x}| > \sigma_i$ (separated by some σ_i by the current classifier). Their outliers removal procedure guarantees the existence of a $\sigma_i > 1/\mathrm{poly}$ that separates at least a polynomial fraction of the distribution. Hence, this step can be applied iteratively until a rule that covers (almost) the whole distribution is produced.

An interesting view of our work is that it reduces the perceptron problem to the *witness identification problem*: for any given bad classifier, find a vector that with high probability constitutes a misclassified example. This view also proved useful in searching for polynomial time algorithm for the problem under various noise functions. We proposed two methods, *slicing* and *scaling*, to handle accuracy issues when producing an estimate of a witness. When using *slices*, our approach also requires the ability to identify when a classifier is good or bad with high probability, with respect to the whole distribution or a large enough (at least 1/poly) portion of it. This additional requirement is not needed when *scaling* is used.

7. Discussion

We suggest some future research directions and point on possible obstacles.

Strong polynomiality. Our algorithm is polynomial in the input size, but not strongly polynomial (since the running time depends on b). This is not surprising since even the noise-free version is easily seen to be equivalent to solving general systems of linear inequalities which is essentially equivalent to solving general Linear Programming. These problems are not known to have a strongly polynomial algorithm. (That is, a strongly polynomial algorithm here is very unlikely since this implies one for general LP.)

Practical polynomial-time algorithm?. An interesting issue is finding an algorithm that has both polynomial-time guarantee and is practical. The Ellipsoid algorithm in itself is not a practical algorithm for LP problems. One approach might be to try to adapt our ideas to work with an interior-point algorithm. Another heuristic approach is to combine the perceptron algorithm, which seems to work well on "real data" with some ideas from our "witness identification" subroutines.

General noise functions $\Delta(\mathbf{x})$. We discuss the tractability of the problem for various *non-uniform* noise models. When the noise is adversarial (for example, $\Delta = 1$ on some regions or points and $\Delta = 0$ or arbitrary elsewhere), the problem is equivalent to the NP-hard problem of finding a solution to a linear system of inequalities that minimizes the number of unsatisfied constraints. (this observation for the perceptron problem was made in [2]).

Our methods do apply when the noise level is non-uniform but provided (that is, along with every sample $\mathbf{x} \in \tilde{\chi}$ we are supplied with the applicable noise rate $\Delta(\mathbf{x})$). In this case, $\Delta(\mathbf{x})$ can be incorporated into our algorithm. Alternatively, our algorithm can be applied as is by increasing the noise level to uniform $\Delta_{\max}$ (when $\Delta(\mathbf{x}) \leq \Delta_{\max}$, for some fixed $\Delta_{\max} < 1/2$). The latter can be done by applying additional noise to each sample $\mathbf{x} \in \tilde{\chi}$ with rate $\Delta' = \frac{\Delta_{\max} - \Delta(\mathbf{x})}{1 - 2\Delta(\mathbf{x})}$ (negate $\mathbf{x}$ it with probability Δ'). (note that examples on which $|\Delta - 0.5|$ is more that polynomially small, have to be disregarded, but our algorithm yields good classification elsewhere.)

In Section 5 we extended our methods to handle noisy data with non-uniform unknown noise when the noise is more highly correlated with proximity to optimal thresholds than with proximity to unsatisfactory thresholds. An intriguing question that is yet unanswered is the existence of a polynomial-time algorithm that handles arbitrary unknown noise, under the constraint that $\forall \mathbf{x}, \Delta(\mathbf{x}) \leq \Delta_{\max} < 1/2$.

References.

[1] S. Agmon. The relaxation method for linear inequalities. *Canadian J. of Mathematics*, 6:382–392, 1954.

[2] E. Amaldi. *From finding maximum feasible subsystems of linear systems to feedforward neural network design.* PhD thesis, Swiss Federal Institute of Technology at Lausanne (EPFL), 1994. (Ph.D. dissertation No. 1282. Department of Mathematics).

[3] J. A. Anderson and E. Rosenfeld, editors. *Neurocomputing: Foundations of research.* MIT Press, 1988.

[4] A. Blum, A. Frieze, R. Kannan, and S. Vempala. A polynomial-time algorithm for learning noisy linear threshold functions. In *Proc. 37th IEEE Annual Symposium on Foundations of Computer Science*, pages 330–338. IEEE, 1996.

[5] T. Bylander. Learning linear threshold functions in the presence of classification noise. In *Proc. 7th Annual ACM conference on Computational Learning Theory*, pages 340–347. ACM, 1994.

[6] T. Bylander. Learning probabilistically consistent linear threshold functions. In *Proc. 10th Annual ACM conference on Computational Learning Theory*, pages 0–000. ACM, 1997.

[7] H. Chernoff. A measure of the asymptotic efficiency for test of a hypothesis based on the sum of observations. *Annals of Math. Statistics*, 23:493–509, 1952.

[8] B. Grötschel, L. Lovasz, and Schrijver A. *Geometric algorithms and combinatorial optimization.* Springer-Verlag, New York, 1988.

[9] J. Hertz, A. Krogh, and R. G. Palmer. *Introduction to the Theory of Neural Computation.* Addison-Wesley Publishing Co., 1991.

[10] M. Kearns. Efficient noise-tolerant learning from statistical queries. In *Proc. 25th Annual ACM Symposium on Theory of Computing*, pages 392–401. ACM, 1993.

[11] M. Kearns and U. Vazirani. *An introduction to computational learning theory.* MIT Press, Cambridge, MA, 1994.

[12] L. G. Khachiyan. A polynomial algorithm in linear programming. *Doklady Akad. Nauk. USSR*, 244:1093–1096, 1979. translated as *Soviet Mathematics Doklady*, **20**, 191–194.

[13] N. Littlestone. *Mistake bounds and logarithmic linear-threshold learning algorithms.* PhD thesis, University of Calif., Santa Cruz, California, 1989.

[14] M. L. Minsky and S. A. Papert. *Perceptrons.* MIT Press, 1969.

[15] R. L. Rivest. Learning decision lists. *Machine Learning*, 2:229–246, 1987.

[16] F. Rosenblatt. *Principles of Neurodynamics.* Spartan Books, 1962.

[17] A. Schrijver. *Theory of Linear and Integer Programming.* John Wiley & Sons, New York, 1987.