On the Difficulty of Approximately Maximizing Agreements

Shai Ben-David and Nadav Eiron
Department of Computer Science
Technion
Haifa 32000, Israel
{shai,nadav}@cs.technion.ac.il

Philip M. Long
Department of Computer Science
National University of Singapore
Singapore 117543, Republic of Singapore
plong@comp.nus.edu.sg

Abstract

We address the computational complexity of learning in the agnostic framework. For a variety of common concept classes we prove that, unless P=NP, there is no polynomial time approximation scheme for finding a member in the class that approximately maximizes the agreement with a given training sample. In particular our results apply to the classes of monomials, axis-aligned hyper-rectangles, closed balls and monotone monomials. For each of these classes we prove the NP-hardness of approximating maximal agreement to within some fixed constant (independent of the sample size and of the dimensionality of the sample space). For the class of half-spaces, we prove that, for any $\epsilon > 0$, it is NP-hard to approximately maximize agreements to within a factor of $(418/415 - \epsilon)$, improving on the best previously known constant for this problem, and using a simpler proof.

An interesting feature of our proofs is that, for each of the classes we discuss, we find patterns of training examples that, while being hard for approximating agreement within that concept class, allow efficient agreement maximization within other concept classes. These results bring up a new aspect of the model selection problem – they imply that the choice of hypothesis class for agnostic learning from among those considered in this paper can drastically effect the computational complexity of the learning process.

1 INTRODUCTION

We study the computational complexity of agnostic learning with a variety of common hypothesis classes. The agnostic framework [14, 19] is a very useful variant of the PAC learning model in which, informally, the learning algorithm is required to do nearly as well as is possible using hypotheses from a given class. Haussler's work [14] (see also [21]) implies that learnability in this model is, in a sense, equivalent to the ability to come up with a member of the hypothesis class that has high *agreement rate* with the training sample, where the agreement rate is the fraction of sample points that the hypothesis classifies correctly.

In this paper we prove that for a variety of hypothesis classes $\mathcal{H}$, including monomials, axis-aligned hyper-rectangles, closed balls, and monotone monomials, there is a constant $\gamma > 1$ for which the following problem is NP-hard:

> *given a set of labeled examples, find a function in $\mathcal{H}$ that has agreement rate within a factor of γ of the best function in the class.*

We also improve on the best previously known constant γ for the class of half-spaces.

It is not hard to see that such a hardness result implies the hardness of finding a hypothesis that has error rate below the error rate of the best hypothesis in the class plus some fixed $\epsilon > 0$ (see [4]).

The hardness of PAC style learning is a very natural question that has been addressed from a variety of viewpoints. It has been shown that, given certain cryptographic assumptions that are stronger than P$\neq$NP or even RP$\neq$NP, classes such as circuits of a constant (but unknown) depth and polynomially many linear threshold gates [18] and AND/OR/NOT gates [20] are hard to learn in the PAC model using any hypotheses. However, such classes are too rich to be considered useful for learning purposes.

Pitt and Valiant [21] showed that it is NP-hard to decide if there is a 2-term DNF that correctly classifies *all* examples in a training sample. Blum and Rivest [6] established a similar result for two-layer linear threshold networks with only two hidden units, and DasGupta, Siegelmann and Sontag [8] extended these results to apply to networks with piecewise linear hidden units.

The earliest hardness results for agnostically learning `simple' classes address the problem of finding a hypothesis that *maximizes* the number of agreements (rather than just approximately maximizing it). Angluin and Laird [2] showed that maximizing agreements with monotone monomials is NP-hard. Kearns and Li [17] established a similar result for general monomials, and Höffgen, Simon and Van Horn [16] for half-spaces.

One may argue, however, that for all practical purposes, a learner may be considered successful if it can produce a hypothesis with accuracy within a small constant factor of the best possible. We are therefore led to the next level of hardness-of-learning results, showing that guarantees of this type cannot be achieved.

Combining a reduction of Kearns and Li [17] with recent results on the hardness of approximating set cover [10] implies that, unless NP $\subseteq$ DTIME($n^{\log\log n}$), finding a monomial that has a ratio of misclassifications within a factor $o(\log n)$ of the minimum is not possible. Arora, Babai, Stern and Sweedyk [3] showed that is NP-hard to minimize disagreements using half-spaces to within any constant factor.[1] Höffgen and Simon [16] established a similar result.

While the work described in the preceding paragraph considers the minimization problem, we show the hardness of the corresponding maximization task. Why prefer one over the other? Note that the task of approximating the optimal solution to within a constant factor emphasizes good performance on different kinds of samples in the two cases. Minimizing disagreements to within a constant factor tends to emphasize performance on clean data, and maximizing agreements to within a constant factor tends to emphasize performance on noisier data. For example, if the best hypothesis in the class agrees with 99% of the sample, an algorithm that minimized disagreements to within a factor of 2 would have to be correct on 98% of the data, where an algorithm that maximized agreement to within a factor of 1.1 would only need to be correct on 90% of the sample. On the other hand, if the best hypothesis got only 80% right, then an algorithm that minimized disagreements to within a factor of 2 would only need to get 60% right, where an algorithm that maximized agreement to within a factor of 1.1 would have to get over 72% right. Also, as mentioned above, if for all inputs, the maximum agreement is at least a constant fraction ξ of the input size,[2] our results imply that minimum disagreement problem cannot be approximated in polynomial time to within an *additive* constant of $\xi(1-1/\gamma)$.

Two previous papers that we know of have addressed the difficulty of maximizing agreement with a sample. Bartlett and Ben-David [4] showed that approximating the agreement ratio of a one-hidden-layer neural network to within a (multiplicative) constant factor that depends on the number of hidden units in the network is NP-hard. Amaldi and Kann [1] showed that approximately maximizing agreements using half-spaces is APX-complete. Also, it is not too hard to see that the following facts

- weak learning implies strong learning [22]
- half-spaces, balls, monomials and axis-aligned rectangles are weak approximators to DNF (see [19])

together imply that a *fully* polynomial time approximation scheme for approximately maximizing agreements using any of half-spaces, balls, monomials or axis-aligned rectangles would imply the learnability of DNF.

In this paper we consider several common concept classes – monomials and monotone monomials over the boolean cube, and half-spaces, balls and hyper-rectangles in Euclidean spaces. For each of these classes, we show that there exists some constant, $\gamma > 1$, such that approximating the optimal agreement ratio in the class to within this constant factor is NP-hard. In the case of half-spaces, we improve on the constant of Amaldi and Kann, and our proof is simpler as well.

Worst-case hardness results in learning theory are often subject to the criticism that the instances witnessing the hardness of the problem are not representative of `real life' instances. Consequently, such results may not reflect the `practical' hardness of the learning task. There is no precise definition for our vague and intuitive notion of `realistic' instances. However, the patterns of instances upon which our hardness proofs rely are `nicely clustered' ensembles. Specifically, for each of the samples used to witness our hardness results there is a ball (either in the Euclidean metric or in the L_∞ metric) that separates the positively labeled points from the negatively labeled ones.

An interesting consequence of our constructions is that for any of the hypothesis classes that we discuss there are input distributions that, while being NP-hard to learn using that class, are efficiently learnable using some of the other hypothesis classes. In particular, there are input patterns that are hard to approximate using rectangles but are easily fully separable by balls, and vice versa. These results have implications to the issue of *model selection*. It is common to discuss the model selection problem from the point of view of information complexity – i.e. the tradeoff between the the ability of a hypothesis in the class to explain the data and generalization ability. Our analyses show that the choice of model class can further influence the computational complexity of finding a good hypothesis, and that a poor choice can simultaneously hurt the algorithm both by failing to explain the data and by making it hard to find a good hypothesis. (The influence of the choice of hypothesis space on the computational complexity of learning in the PAC model was addressed by Pitt and Valiant [21].)

There is yet another implication of our results to model selection issues. Once a learner sees the training data, it would be desirable to be able to choose the hypothesis class, relative to which the learning process will proceed, so as to maximize the fit between the class and the data. We show that the task of estimating the agreement rate of the optimal hypothesis in any of the above mentioned classes (to within a constant factor) is NP-hard. It follows that the task of choosing a class to work with for a given data is in itself computationally difficult.

The paper is organized as follows: after providing the basic definitions in Section 2, Section 3 lists our computational hardness results. Sections 4 and 5 are where we prove these hardness results. In Section 6 we discuss the implications of our results on the computational aspects of model selection.

2 PRELIMINARIES

In this work, we consider several maximization problems. A maximization problem Π associates with any possible input I a set of feasible solutions C_I. Each input and feasible solution pair is associated a profit denoted $q_\Pi(c, I) \in \mathbf{R}$. We denote the optimal profit of a maximization problem as follows:

$$\mathrm{opt}_\Pi(I) = \max_{c \in C_I} q_\Pi(c, I).$$

Definition 1 *A γ-approximation algorithm A for the maximization problem Π is an algorithm that on any input I out-*

[1]With stronger complexity theoretic assumptions, they could prove stronger statements.

[2]For all the classes considered in this work $\xi \geq 1/2$.

puts a feasible solution $A(I) \in C_I$ such that:

$$\gamma q_\Pi(A(I), I) \geq opt_\Pi(I).$$

Naturally, this definition is meaningful for $\gamma > 1$.

This work concerns maximum agreement problems.

Definition 2 *For a domain set D and a class $\mathcal{C}$ of functions from D to $\{0,1\}$, we define the* Maximum Agreement *problem $\mathcal{C}$-MA as follows: The input of the problem consists of a* sample *S, which is a finite multi-set of elements from $D \times \{0,1\}$. The first component of an example is called its* point*, while the second component of the example is called its* label*. The set $\mathcal{C}$ is the set of feasible solutions. We say some function $f \in \mathcal{C}$ agrees with an example (x, y) if $f(x) = y$. The profit function q for a Maximum Agreement problem is simply the number of examples in S with which the solution agrees.*

We will many times view the elements of $\mathcal{C}$ as subsets of D, rather than as functions from D to $\{0,1\}$. We say a subset $C \subseteq D$ agrees with an example (x, y) if its characteristic function agrees with (x, y).

The first type of maximum agreement problems we consider are problems based on boolean formulas. We shall consider the $\mathcal{C}$-MA problems for the following boolean classes $\mathcal{C}$:

Monomials A monomial over the variables $w_1, \ldots, w_n$ is a conjunction of some literals defined over these variables. We will denote the Monomials-MA problem as MMA.

Monotone Monomials A monotone monomial is simply a monomial that includes only positive literals. We will denote the MonotoneMonomials-MA problem as MMMA.

Anti-Monotone Monomial An anti-monotone monomial is a monomial that includes only negative literals. We will denote the AntiMonotoneMonomials-MA problem as AMMA.

The second type of maximum agreement problems we consider are problems where the definition of the class $\mathcal{C}$ is based on geometric concepts. We shall consider the $\mathcal{C}$-MA problems for the following geometric classes $\mathcal{C}$:

Closed Balls A ball G in $\mathbf{R}^n$ is represented by $\vec{w} \in \mathbf{R}^n$ and $\theta \in R$ such that $G = \{\vec{x} : d(\vec{w}, \vec{x}) \leq \theta\}$. We will denote the Balls-MA problem as BMA.

Half-spaces A half-space H in $\mathbf{R}^n$ is represented by $\vec{w} \in \mathbf{R}^n$ and $\theta \in \mathbf{R}$ and defined to be: $H = \{\vec{x} : \vec{w} \cdot \vec{x} \geq \theta\}$. We will denote the Half-Space-MA problem as HMA.

Hyper Rectangles An Axis Aligned Hyper-rectangle R is represented by $\vec{r}, \vec{s} \in \mathbf{R}^n$ where $R = \prod_{i=1}^{n} [r_i, s_i]$. We will denote the Hyper-Rectangle-MA problem as RMA.

Our basic hardness results will use reductions from MAX-E2-SAT. MAX-E2-SAT is defined as follows: Given as input a set of 2-clauses (each clause is a disjunction of two literals) over a set of n variables $w_1, \ldots, w_n$, find an assignment $\vec{x} \in \{0,1\}^n$ such that the number of clauses that are satisfied is maximized. We will denote the profit function for MAX-E2-SAT by q_{ME2S}. The following theorem, due to Håstad [13], shows that approximating the optimal solution to MAX-E2-SAT is hard.

Theorem 3 *Assuming P≠NP, there is no polynomial time $(22/21 - \epsilon)$-approximation algorithm for MAX-E2-SAT, for any $\epsilon > 0$.*

3 MAIN RESULTS

To show the hardness of the various maximum agreement problems presented above, we use two basic reductions from MAX-E2-SAT: One to Monomial Maximum Agreement, and the other to Ball Maximum Agreement. These two reductions, along with Theorem 3, allow us to prove the following two theorems:

Theorem 4 *If NP≠P, then for any $\epsilon > 0$ there is no polynomial time $(770/767 - \epsilon)$-approximation algorithm for Monomial Maximum Agreement.*

Theorem 5 *If NP≠P, then for any $\epsilon > 0$ there is no polynomial time $(418/415 - \epsilon)$-approximation algorithm for Ball Maximum Agreement.*

Since the set of restrictions of indicator functions for axis-aligned rectangles to $\{0,1\}^n$ are exactly the indicator functions for monomials, Theorem 4 directly implies the following.

Theorem 6 *If NP≠P, then for any $\epsilon > 0$ there is no polynomial time $(770/767 - \epsilon)$-approximation algorithm for Axis-Aligned Rectangle Maximum Agreement.*

Finally, we also show how the construction used in the proof of Theorem 4 may also be used to prove the following:

Theorem 7 *If NP≠P, then for any $\epsilon > 0$ there is no polynomial time $(770/767 - \epsilon)$-approximation algorithm for Monotone Monomial Maximum Agreement.*

A variation of the reduction used for the Balls Maximum Agreement problem (utilizing the same gadget construction) is used to prove the theorem on hardness of the Half-space Maximum Agreement problem. This problem was shown to be impossible to approximate in polynomial time to within $462/461 - \epsilon$ by Amaldi and Kann [1]. We improve upon their hardness result by proving the following:

Theorem 8 *If NP≠P, then for any $\epsilon > 0$ there is no polynomial time $(418/415 - \epsilon)$-approximation algorithm for Half-space Maximum Agreement.*

In Section 6 we provide some further initial results on the influence model selection has on the computational complexity of learning.

4 MONOMIALS AND AXIS-ALIGNED HYPER-RECTANGLES

In this section, we establish the hardness results for Monomials and Axis-Aligned Hyper-rectangles.

4.1 GADGET CONSTRUCTION FOR MONOMIALS

In the instance transformation, for each of the $2n$ possible *literals* in an instance I of MAX-E2-SAT, there will be a *variable* in the instance $f(I)$ of MMA. For each literal ℓ over a variable in I, let w_ℓ be the corresponding variable in $f(I)$.

For each clause C, if ℓ_1 and ℓ_2 are the literals in C, define $\phi(C)$ to consist of five examples as follows:

- The first two examples are labeled 1 and each of their points sets exactly one variable to be true. The first of these examples sets only w_{ℓ_1} to be true, and the second sets only w_{ℓ_2} to be true.
- The next two examples are labeled 0, and their points set exactly two variables to be true. The first of these examples sets only w_{ℓ_1} and $w_{\bar{\ell}_1}$ to be true, and the second sets only w_{ℓ_2} and $w_{\bar{\ell}_2}$ to be true.
- The last example is also labeled 0, and its point also sets exactly two variables to be true: w_{ℓ_1} and w_{ℓ_2}.

For some instance I of MAX-E2-SAT with m clauses, $f(I)$ consists of all examples in any $\phi(C)$ for a clause C in I, together with $5m$ copies of an example labeled 1 whose point does not set any variable to true.

For the solution transformation g, given some monomial M the assignment $g(M)$ is defined as follows: each variable v_i is set to "true" if and only if $\bar{w}_{v_i}$ does not appear in M.

4.2 ANALYZING THE REDUCTION

Fix an algorithm B for maximizing agreements using monomials, and an arbitrary instance I of MAX-E2-SAT. Let n be the number of variables and m be the number of clauses in I. Let M be the monomial output by B on input $f(I)$. Let $\mathrm{opt}_{\mathrm{MMA}}(f(I))$ be the maximum number of examples in $f(I)$ that any monomial agrees with. Assume that $\gamma \in (0, 3/7)$ satisfies

$$q_{\mathrm{MMA}}(M, f(I)) \geq (1-\gamma)\mathrm{opt}_{\mathrm{MMA}}(f(I)).$$

Let $a = g(M)$ be the assignment output by A.

Lemma 9 $\mathrm{opt}_{\mathrm{MMA}}(f(I)) \geq \mathrm{opt}_{\mathrm{ME2S}}(I) + 8m.$

Proof: Let a^* be an optimal assignment for I. Define M^* to be $\wedge_\ell \bar{w}_\ell$, where ℓ ranges over those literals not satisfied by a^*.

First, since M^* has only negated literals, it is satisfied by the assignment that sets all variables to false, and therefore correctly classifies the $5m$ examples where this point is labeled 1. Second, since for each literal ℓ using a variable in I, either $\bar{w}_\ell$ or $\bar{w}_{\bar{\ell}}$ is included in M^*, all of the points in which these two components are both set to true do not satisfy M^*, and therefore are classified correctly.

Choose a clause C, and let ℓ_1 and ℓ_2 be the literals in C.

Suppose that a^* satisfies exactly one of the literals in C, and assume without loss of generality that it is ℓ_1. Then the point in $\phi(C)$ that sets only w_{ℓ_1} to true satisfies M^*, and the point that sets w_{ℓ_1} and w_{ℓ_2} to true does not satisfy M^*, so both of these examples of $\phi(C)$ are classified correctly by M^*.

If both literals in C are satisfied by a^*, then both points in $\phi(C)$ which set the single variables w_{ℓ_1} and w_{ℓ_2} respectively to true satisfy M^*, and they are both classified correctly.

If a^* doesn't satisfy a clause C, then M^* does not contain the point in $\phi(C)$ that sets exactly w_{ℓ_1} and w_{ℓ_2} to true, and therefore classifies this point correctly.

So if a^* satisfies C, 4 of the examples in $\phi(C)$ are classified correctly by M^*, otherwise 3 are. Adding the $5m$ copies of the assignment that sets all variables to false completes the proof. ∎

Lemma 10 *M does not contain any unnegated literals.*

Proof: We have

$$\begin{aligned} q_{\mathrm{MMA}}(M, f(I)) &\geq (1-\gamma)\mathrm{opt}_{\mathrm{MMA}}(f(I)) \\ &\geq (1-\gamma)(\mathrm{opt}_{\mathrm{ME2S}}(I) + 8m) \\ &\qquad \text{(by Lemma 9)} \\ &\geq (1-\gamma)(3/4m + 8m) \\ &> 5m \end{aligned}$$

since $\gamma < 3/7$. Therefore, since there are a total of $10m$ examples in $f(I)$, M must agree with the $5m$ examples in which the assignment which sets all variables to false is labeled with 1, completing the proof. ∎

Lemma 11 *For any clause C in I, if a does not satisfy C, then M agrees with at most 3 of the examples in $\phi(C)$.*

Proof: Choose one of the literals in C, and let i be its variable. If $v_i \in C$, then $\bar{w}_{v_i}$ is included in M, and so M disagrees with the example in which the assignment which only sets w_{v_i} to true is labeled 1. If $\bar{v}_i \in C$, then $\bar{w}_{v_i}$ is not included in M, and so

- if $\bar{w}_{\bar{v}_i}$ is not included in M then M disagrees with the example in which the point which only sets w_{v_i} and $w_{\bar{v}_i}$ to true is labeled 0, and
- if $\bar{w}_{\bar{v}_i}$ is included in M, then M disagrees with the example in which the point which only sets $w_{\bar{v}_i}$ to true is labeled 1.

So M disagrees with at least one example for each of the literals in C, completing the proof. ∎

Lemma 12 *For any clause C in I, M agrees with at most 4 of the examples in $\phi(C)$.*

Proof: If M agrees with both of the examples that are labeled 1, then it must disagree with the example in which the point where both variables corresponding to literals in C are set to true is labeled 0. ∎

Now we are ready to sum up the analysis of this reduction.

Lemma 13

$$q_{\mathrm{ME2S}}(a, I) \geq (1 - 35\gamma/3)\mathrm{opt}_{\mathrm{ME2S}}(I).$$

Proof: We have

$$
\begin{aligned}
q_{\mathrm{ME2S}}(a, I) &\geq q_{\mathrm{MMA}}(M, f(I)) - 8m \\
&\qquad\qquad \text{(by Lemmas 11 and 12)} \\
&\geq (1-\gamma)\mathrm{opt}_{\mathrm{MMA}}(f(I)) - 8m \\
&\qquad\qquad \text{(by assumption)} \\
&\geq (1-\gamma)(\mathrm{opt}_{\mathrm{ME2S}}(I) + 8m) - 8m \\
&\qquad\qquad \text{(by Lemma 9)} \\
&= (1-\gamma)\mathrm{opt}_{\mathrm{ME2S}}(I) - 8\gamma m \\
&\geq (1-\gamma)\mathrm{opt}_{\mathrm{ME2S}}(I) - (32/3)\gamma\ \mathrm{opt}_{\mathrm{ME2S}}(I) \\
&\qquad \text{(since } \mathrm{opt}_{\mathrm{ME2S}}(I) \geq (3/4)m\text{)} \\
&= (1 - 35\gamma/3)\mathrm{opt}_{\mathrm{ME2S}}(I),
\end{aligned}
$$

completing the proof. ∎

This, combined with Theorem 3, immediately proves Theorem 4.

4.3 A HARDNESS RESULT FOR MONOTONE MONOMIALS

Since, by Lemma 10, the optimal solution (as well as any approximate solution with cost not less than $4/7$ of the optimal) contains only negated literals, the same hardness result applies to classification by anti-monotone monomials (monomials with all literals negated). We thus have the following:

Lemma 14 *Assume B is a polynomial time γ-approximation algorithm for MMMA. Then AMMA can be γ-approximated in polynomial time.*

Proof: Given an instance I of AMMA, run B with input $\overline{I}$ where $\overline{I}$ is constructed from I by flipping every component in every sample point between 1 and 0. Given the monotone monomial $M = B(\overline{I})$, output the monomial M' which is constructed from M by negating every literal. It is immediate that the profit M' achieves on I is the same as the profit M achieves on $\overline{I}$, and that M' is indeed anti-monotone. Additionally, the optimal solution to AMMA with input I has the same profit as the optimal solution to MMMA with input $\overline{I}$. ∎

This last Lemma, along with Lemma 10 and Theorem 4 immediately prove Theorem 7.

5 BALLS AND HALF-SPACES

Next, we turn to the problems of Ball Maximum Agreement and Half-space Maximum Agreement. The reductions for both these problems rely on the same basic construction, detailed below.

5.1 GADGET CONSTRUCTION FOR BALLS AND HALF-SPACES

Choose an algorithm B for the problem of maximizing agreements using balls. We will use B to construct an algorithm A for MAX-E2-SAT using a reduction. That is, given an instance I of MAX-E2-SAT, A will construct a sample $f(I)$, and pass it to algorithm B. After B outputs a ball G, A will construct an assignment $g(G)$, and output it.

In the instance transformation f, corresponding to each variable in the instance I of MAX-E2-SAT, there will be one

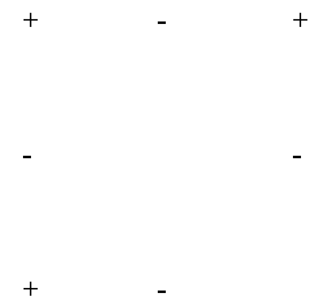

Figure 1: One possible value of the examples in $\phi(C)$ of the gadget used for Balls and Half-spaces: only the components with nonzero values, i.e. those that correspond to variables in C, are shown. Examples whose label is 1 are plotted with a "+", and those whose label is 0 are plotted with a "-".

component of the points in the examples of the sample $f(I)$. Let us suppose there are a total of m clauses in I, which use a total of n variables, and suppose the variables are $v_1, ..., v_n$. Associated with each clause C in I, there will be a collection $\phi(C)$ of examples in $f(I)$. All of the points in examples in $\phi(C)$ will have nonzero entries only in components that correspond to variables appearing in C. In Figure 1, the projections of the examples in one possible value of $\phi(C)$ onto the two components corresponding to the variables in C are plotted. The "missing +" is in the position corresponding to the assignment to the two variables in C that fails to satisfy C.

Now, let us specify ϕ formally. For each clause C and each variable v_j from I, define

$$
\psi_j(C) = \begin{cases} 1 & \text{if } v_j \in C \\ -1 & \text{if } \overline{v}_j \in C \\ 0 & \text{otherwise.} \end{cases}
$$

Then $\phi(C)$ consists of

- An example consisting of the point $\vec{u}_C = (\psi_1(C), ..., \psi_n(C))$ and the label 1.
- Two more examples whose label is 1, and each of whose points is obtained by negating one of the nonzero components of $\vec{u}_C$.
- Two examples whose label is 0, and each of whose points is obtained by zeroing out one of the nonzero components of $\vec{u}_C$.
- Two more examples whose label is 0, and whose points are obtained by negating the single nonzero components of the two 0-labeled examples described in the previous bullet point.

If ϕ is defined this way, then $f(I)$ is the multi-set consisting of all examples in $\phi(C)$ for all clauses C in I.

Now we turn to the solution transformation g. Choose some ball G, specified by a center point $\vec{w}$ and a radius θ. Then let $g(G)$ be the assignment in which the ith variable is set to "true" if and only if w_i is nonnegative.

We use the exact same construction to show the reduction to Half-space Maximum Agreement. The only part that changes is the solution transformation g. Given a half-space H, specified by a vector $\vec{w} \in \mathbf{R}^n$, and a threshold $\theta \in \mathbf{R}$. $g(H)$ will then be the assignment in which the variable v_i is set to "true" if and only if w_i is nonnegative.

5.2 ANALYSIS OF THE BALLS REDUCTION

Fix an arbitrary instance I of MAX-E2-SAT. Let n be the number of variables and m be the number of clauses. Let G be the ball output by an algorithm B on input $f(I)$, and let $\vec{x} \in \mathbf{R}^n$ and $\theta \in \mathbf{Q}$ be its representation. Let $a = g(G)$ be the assignment output by A.

Lemma 15 *For any clause C in I, if a does not satisfy C, then G agrees with at most 4 of the examples in $\phi(C)$.*

Proof: Assume without loss of generality that the first two variables appear in C, and suppose that $C = \{v_1, v_2\}$ (the other cases can be handled similarly). Since a does not satisfy C, the definition of g implies that the first and second coordinates of $\vec{w}$, the center of G, are non-positive. We claim the following hold:

(a) If $(1, -1, 0, ..., 0) \in G$, then $(0, -1, 0, ..., 0) \in G$ (recall that the first of those is labeled 1 in $\phi(C)$, and the second is labeled 0).

(b) If $(-1, 1, 0, ..., 0) \in G$, then $(-1, 0, 0, ..., 0) \in G$.

(c) If $(1, 1, 0, ..., 0) \in G$, then both $(1, 0, 0, ..., 0)$ and $(0, 1, 0, ..., 0)$ are in G.

To see that this is indeed the case, simply check that $d((1, -1, 0, ..., 0), \vec{w}) > d((0, -1, 0, ..., 0), \vec{w})$ for any point $\vec{w}$ whose first two coordinates are non-positive. Hence, any ball centered in such a point that contains $(1, -1, 0, ..., 0)$ must contain $(0, -1, 0, ..., 0)$ as well. The same can be easily verified for (b) and (c). $\square$

Lemma 16 *For any clause C in I, G agrees with at most 5 of the examples in $\phi(C)$.*

Proof: Clearly, if G is to agree with more than 4 points from $\phi(C)$, it must include at least one point that is labeled positive. Assume again, w.l.o.g., that $C = \{v_1, v_2\}$. If both the points $(1, -1, 0, \ldots, 0)$ and $(1, 1, 0, \ldots, 0)$ are in G, then $(1, 0, 0, \ldots, 0)$ must also be in G, as G is convex. Also, if both $(-1, 1, 0, \ldots, 0)$ and $(1, 1, 0, \ldots, 0)$ are in G, then $(0, 1, 0, \ldots, 0)$ must also be in G. Thus, if G includes two positive points, it must include at least one negative point. Finally, note that if it includes all 3 positive points, it must include at least two negative points. Thus, no more than 5 points are correctly classified. $\square$

Let $\mathrm{opt}_{\mathrm{ME2S}}(I)$ be the maximum number of clauses in I than can be satisfied by any truth assignment, and let $\mathrm{opt}_{\mathrm{BMA}}(f(I))$ be the maximum number of examples in $f(I)$ that any ball agrees with.

Lemma 17 $\mathrm{opt}_{\mathrm{BMA}}(f(I)) \geq \mathrm{opt}_{\mathrm{ME2S}}(I) + 4m$

Proof: Let a^* be the optimal assignment for I. Define $\vec{w}^*$ by

$$w_i^* = \begin{cases} 1 & \text{if } a^*(v_i) = \text{true} \\ -1 & \text{if } a^*(v_i) = \text{false}. \end{cases}$$

Let $\theta^* = \sqrt{n-2}$, $G^* = \{\vec{x} : d(\vec{x}, \vec{w}^*) \leq \theta^*\}$, and C be some clause. Since each point $\vec{u} \in \phi(C)$ that is labeled 0 has exactly one non-zero coordinate, and all the coordinates of $\vec{w}^*$ are either 1 or -1, we have $d(\vec{u}, \vec{w}^*) \geq \sqrt{n-1}$. Therefore, G^* agrees with all 0 labeled points in $\phi(C)$. On the other hand, if C is satisfied by a^*, then the point $\vec{u}$ that has two coordinates set to the same sign and value as in $\vec{w}^*$ is labeled 1 in $\phi(C)$. Hence, we have $d(\vec{u}, \vec{w}^*) = \sqrt{n-2}$ and $u \in G^*$.

Thus, for each clause C that is satisfied by a^*, H^* agrees with at least 5 examples in $\phi(C)$, and for each clause C that is not satisfied by a^*, H^* agrees with at least 4 of the examples in $\phi(C)$. This completes the proof. $\square$

Now we are ready to sum up the analysis of our reduction

Lemma 18 *If*

$$q_{\mathrm{BMA}}(G, f(I)) \geq (1-\gamma)\mathrm{opt}_{\mathrm{BMA}}(f(I))$$

then

$$q_{\mathrm{ME2S}}(a, I) \geq (1 - 19\gamma/3)\mathrm{opt}_{\mathrm{ME2S}}(I).$$

Proof: We have

$$\begin{aligned} q(a, I) &\geq q(G, f(I)) - 4m \\ &\qquad \text{(by Lemmas 15 and 16)} \\ &\geq (1-\gamma)\mathrm{opt}_{\mathrm{BMA}}(f(I)) - 4m \\ &\qquad \text{(by assumption)} \\ &\geq (1-\gamma)(\mathrm{opt}_{\mathrm{ME2S}}(I) + 4m) - 4m \\ &\qquad \text{(by Lemma 17)} \\ &= (1-\gamma)\mathrm{opt}_{\mathrm{ME2S}}(I) - 4\gamma m \\ &\geq (1-\gamma)\mathrm{opt}_{\mathrm{ME2S}}(I) - (16/3)\gamma\, \mathrm{opt}_{\mathrm{ME2S}}(I) \\ &\qquad \text{(since } \mathrm{opt}_{\mathrm{ME2S}}(I) \geq (3/4)m) \\ &= (1 - 19\gamma/3)\mathrm{opt}_{\mathrm{ME2S}}(I), \end{aligned}$$

completing the proof. $\square$

Using this last Lemma, and Theorem 3, and doing some calculations proves Theorem 5.

5.3 ANALYSIS OF THE HALF-SPACE REDUCTION

The analysis of the reduction for the HMA problem is very similar to that of the BMA problem.

Fix I, n and m as for the analysis of the balls reduction. Let H be the half-space output by B on input $f(I)$, and let $\vec{w} \in \mathbf{R}^n$ and θ be its representation, i.e. $H = \{\vec{x} : \vec{w} \cdot \vec{x} \geq \theta\}$. Let $a = g(H)$ be the assignment output by A.

Lemma 19 *For any clause C in I, if a does not satisfy C, then H agrees with at most 4 of the examples in $\phi(C)$.*

Proof: Assume without loss of generality that the first two variables appear in C, and suppose that $C = \{v_1, v_2\}$ (the other cases can be handled similarly). We claim that the following hold:

(a) If $(1, -1, 0, ..., 0) \in H$, then $(0, -1, 0, ..., 0) \in H$ (recall that the first of those is labeled 1 in $\phi(C)$, and the second is labeled 0).

(b) If $(-1, 1, 0, ..., 0) \in H$, then $(-1, 0, 0, ..., 0) \in H$.

(c) If $(1,1,0,...,0) \in H$, then both $(1,0,0,...,0)$ and $(0,1,0,...,0)$ are in H.

Taking (a), (b), and (c) together, at least as many points that are labeled 0 in $\phi(C)$ as are labeled 1 are contained in H. Since overall $\phi(C)$ has one more point labeled 0 than labeled 1, this implies that the number of points labeled 0 that are not in H is at most one more than the number of points labeled 1 that are not in H. Therefore, overall, H agrees with at most one more point than it disagrees with, which implies that H agrees with at most 4 points in $\phi(C)$. So if we can prove (a), (b), and (c), we are done.

Since a doesn't satisfy C, it must set v_1 and v_2 to "false", which implies w_1 and w_2 are negative.

To prove (a), note that if H contains $(1,-1,0,...,0)$, then $w_1 - w_2 \geq \theta$. But since $w_1 < 0$, this implies $-w_2 \geq \theta$, which means $(0,-1,0,...,0) \in H$. One can prove (b) similarly.

If $(1,1,0,...,0) \in H$, then $w_1 + w_2 \geq \theta$, but since w_1 and w_2 are negative, then $w_1 \geq \theta$ and $w_2 \geq \theta$, proving (c) and completing the proof. ∎

A nearly identical proof, which is omitted, establishes the following. (This lemma can alternatively be verified with a trivial case analysis.)

Lemma 20 *For any clause C in I, H agrees with at most 5 of the examples in $\phi(C)$.*

Let $\mathrm{opt}_{\mathrm{ME2S}}(I)$ be the maximum number of clauses in I than can be satisfied by any truth assignment, and let $\mathrm{opt}_{\mathrm{HMA}}(f(I))$ be the maximum number of examples in $f(I)$ that any half-space agrees with.

Lemma 21 $\mathrm{opt}_{\mathrm{HMA}}(f(I)) \geq \mathrm{opt}_{\mathrm{ME2S}}(I) + 4m$

Proof: Let a^* be the optimal assignment for I. Define $\vec{w}^*$ by

$$w_i^* = \begin{cases} 1 & \text{if } a^*(v_i) = \text{true} \\ -1 & \text{if } a^*(v_i) = \text{false}. \end{cases}$$

Let $\theta^* = 2$, $H^* = \{\vec{x} : \vec{w}^* \cdot \vec{x} \geq \theta^*\}$ and C be some clause. Since each point in $\phi(C)$ that is labeled 0 has a single nonzero component in $\{-1,1\}$, it cannot be in H^*. Therefore H^* agrees with all the examples in $\phi(C)$ that are labeled 0. Since each point $\vec{u}$ that is labeled 1 in $\phi(C)$ has two nonzero components, and they are in $\{-1,1\}$, $\vec{w}^* \cdot \vec{u} \geq 2$ if and only if both of these have the same sign as the corresponding components of $\vec{w}^*$. If a^* satisfies C, then the definition of $\phi(C)$ implies that one such point is labeled 1 in $\phi(C)$.

Thus, for each clause C that is satisfied by a^*, H^* agrees with at least 5 examples in $\phi(C)$, and for each clause C that is not satisfied by a^*, H^* agrees with at least 4 of the examples in $\phi(C)$. This completes the proof. ∎

Now we are ready to sum up the analysis of our reduction

Lemma 22 *If*

$$q_{\mathrm{HMA}}(H, f(I)) \geq (1-\gamma)\mathrm{opt}_{\mathrm{HMA}}(f(I))$$

then

$$q_{\mathrm{ME2S}}(a, I) \geq (1 - 19\gamma/3)\mathrm{opt}_{\mathrm{ME2S}}(I).$$

Proof: We have

$$\begin{aligned}
q(a,I) &\geq q(H, f(I)) - 4m \\
&\qquad \text{(by Lemmas 19 and 20)} \\
&\geq (1-\gamma)\mathrm{opt}_{\mathrm{HMA}}(f(I)) - 4m \\
&\qquad \text{(by assumption)} \\
&\geq (1-\gamma)(\mathrm{opt}_{\mathrm{ME2S}}(I) + 4m) - 4m \\
&\qquad \text{(by Lemma 21)} \\
&= (1-\gamma)\mathrm{opt}_{\mathrm{ME2S}}(I) - 4\gamma m \\
&\geq (1-\gamma)\mathrm{opt}_{\mathrm{ME2S}}(I) - (16/3)\gamma\ \mathrm{opt}_{\mathrm{ME2S}}(I) \\
&\qquad \text{(since } \mathrm{opt}_{\mathrm{ME2S}}(I) \geq (3/4)m\text{)} \\
&= (1 - 19\gamma/3)\mathrm{opt}_{\mathrm{ME2S}}(I),
\end{aligned}$$

completing the proof. ∎

Using this last Lemma, and Theorem 3, and doing some calculations proves Theorem 8.

6 COMPUTATIONAL ASPECTS OF MODEL SELECTION

The constructions we used for our reductions illustrate an aspect of model selection related to the computational complexity of learning: While our constructions yield data sets that are computationaly hard to learn using one concept classes, they are easy to learn using other concept classes. More generally, there exists, for each of the concept classes we considered above, a subset of its legal inputs for which the maximum agreement problem can be solved in polynomial time. On the other hand, these subsets, of "easy" inputs for one concept class, include inputs for which maximizing agreements using the other concept classes we consider is NP-hard. Therefore, when considering the problem of model selection, one must take into account not only the approximation error of a class, but also the computational complexity of using that class with the data at hand[3].

We now turn to listing the conditions under which the various Maximum Agreement problems we consider become easy to solve:

Claim 23 *([7]) The RMA problem may be solved in polynomial time for inputs that are separable by an axis-aligned rectangle.*

Claim 24 *The BMA problem for inputs that satisfy:*

1. *There exists a ball of radius 1 that contains* all *the positive points and none of the negative points of the input.*
2. *All negative points are at distance at least $1 + c$ (for some constant c) from the center of all balls of radius 1 that captures all positive points.*

is solvable in polynomial time.

[3] In the examples we show here, the two considerations, namely approximation error and computational complexity, happen to coincide: exact fitting goes hand in hand with low computational complexity. The question remains whether this is true in general, though.

In the proof of this claim, we use the following result of Ben-David et al. [5]:

Theorem 25 *[5, Theorem 5.1] There exists a family $(A_k)_{k\geq 1}$ of polynomial time algorithms such that the following holds: For all $n \geq 2$ and $k \geq 1$, A_k on input $S \subseteq \Re^n$ outputs a point $y \in \Re^n$ such that the closed ball $\bar{B}(y, 1+\sqrt{1/k})$ contains not less points of S than the optimal closed ball of radius 1.*

We are now ready to sketch the proof of Claim 24:

Proof Sketch: We show an algorithm to solve the Ball Maximum Agreement problem for such inputs, based on the unsupervised learning algorithm for balls of [5]. We use this algorithm in the following way: We strip the original input of all the negative points, and pass as input to the algorithm of Ben-David et al. all the positive points. We receive as output a ball of radius $1+\sqrt{1/k}$, which we use as the output to the Maximum Agreement problem.

It is easy to see that, on the kinds of inputs we consider, if we pick k so that $\sqrt{1/k} \leq c/2$, the resulting ball will include all positive points and none of the negative ones, hence solving the Maximum Agreement problem exactly. ∎

Corollary 26 *For the set of labeled inputs that are separable by an axis-aligned hyper-rectangle, the Axis-Aligned Hyper-Rectangle Maximum Agreement problem may be solved in polynomial time, yet the Ball Maximum Agreement problem cannot be approximated for these inputs, to within $418/415-\epsilon$, in polynomial time (for any $\epsilon > 0$), unless P=NP.*

Proof: Recall the construction of sample points used in Section 5.1. It can be easily verified that, by slightly increasing the non-zero component of the 0-labeled points (from 1 to, say $\sqrt{2}$), the reduction remains valid by modifying the proof of Lemma 15. The resulting construction is clearly separable by rectangles, since the rectangle defined by the interval $[-1, 1]$ in all coordinates would include only 1-labeled points. Therefore, by Claim 23, a separating rectangle may be found in polynomial time. ∎

Furthermore, by making a slight modification to the construction, we can show the following:

Corollary 27 *For the set of labeled inputs satisfying the conditions of Claim 24, the Closed Ball Maximum Agreement problem may be solved in polynomial time, yet the Open Half-space Maximum Agreement problem[4] cannot be approximated for these inputs, to within $418/415-\epsilon$, in polynomial time (for any $\epsilon > 0$), unless P=NP.*

Proof: We consider the construction given in Section 5.1, but with the labeling reversed. Now, all positive points are at a distance of 1 from the origin, while all negative points are at a distance of $\sqrt{2}$ from the origin. This input clearly satisfies the conditions of Claim 24 with $c = \sqrt{2}-1$, and hence a separating ball may be found in polynomial time. On the other hand, the complement of a closed half-space is an open half-space. Therefore, it is immediate to see that this new gadget is hard to approximate to within $418/415-\epsilon$ using open half-spaces. ∎

[4]By open half-space we mean that the half-space is defined as $H = \{\vec{x} : \vec{w}\cdot\vec{x} > \theta\}$.

On the other hand, we can also show a construction that is easy for balls to classify, and hard for rectangles to approximate. The construction we used for Monotone Monomials (which is thus also hard for rectangles to classify correctly) is very easy for balls: Note that all points labeled 1 have at most one coordinate that is set to 1, while all 0 labeled points have at least two coordinates set to 1. By adding a single positive point of the form $(-1, 0, \ldots, 0)$, the input would satisfy the conditions of Claim 24 with $c = \sqrt{2}-1$, making it easy to separate with a ball. On the other hand, it is easy to see that the hardness result for hyper-rectangles remain essentially unchanged. Hence we have:

Corollary 28 *For the set of labeled inputs satisfying the conditions of Claim 24, the Closed Ball Maximum Agreement problem may be solved in polynomial time, yet the Axis Aligned Rectangle Maximum Agreement problem cannot be approximated for these inputs, to within $770/767-\epsilon$, in polynomial time (for any $\epsilon > 0$), unless P=NP.*

We therefore see that the same sample may be computationally easy to learn using one hypotheses class, while being NP-hard to learn using a different class. Furthermore, these is no clear hierarchy in the power of these hypotheses classes: While balls are superior to rectangles on some inputs, rectangles are superior to balls on other inputs.

While our proofs, as they appear above, show the hardness of actually *finding* a solution to the approximation problem, we also claim that approximating the optimal *agreement rate* for these problems is as hard as finding the optimal solution. Clearly, finding a solution is at least as hard as finding its profit. To see that the converse is also true, note that all the reductions we used in our construction would work just the same if all we wanted was to estimate the profit of the optimal solution. By examining Håstad's proof for the hardness of approximating MAX-E2-SAT, one can verify that the result still holds even if all that is required is to approximate the number of clauses that may be satisfied. Thus, we have the following results:

Corollary 29 *It is NP-hard to approximate the optimal agreement rate for the Monomial Maximum Agreement problem, to within a factor of $(770/767-\epsilon)$, for any $\epsilon > 0$..*

Corollary 30 *It is NP-hard to approximate the optimal agreement rate for the Ball Maximum agreement problem, to within a factor of $(418/415-\epsilon)$ for any $\epsilon > 0$.*

Corollary 31 *It is NP-hard to approximate the optimal agreement rate for the Axis-Aligned Rectangle Maximum Agreement problem, to within a factor of $(770/767-\epsilon)$, for any $\epsilon > 0$.*

Corollary 32 *It is NP-hard to approximate the optimal agreement rate for the Monotone Monomial Maximum Agreement problem, to within a factor of $(770/767-\epsilon)$, for any $\epsilon > 0$.*

Corollary 33 *It is NP-hard to approximate the optimal agreement rate for the Half-space Maximum agreement problem, to within a factor of $(418/415-\epsilon)$ for any $\epsilon > 0$.*

7 CONCLUSION

In this paper, we have established the hardness of approximately maximizing agreement with a sample using a variety of simple hypothesis classes, and discussed consequences of our proofs concerning the problem of model selection.

We do not know of any nontrivial approximation algorithms for the problems addressed in this paper; the emphasis of this critereon on performance on dirty samples suggests that this framework may be useful for evaluating algorithms for learning using simple hypotheses. Such algorithms could then potentially be applied in practice in conjunction with boosting [12, 11]. Ben-David, Eiron and Simon [5] consider a different notion of approximation for densest region detection. Their results may be used to construct efficient algorithms for the maximum agreement problem for some of the classes we consider. Given a margin parameter μ, these algorithms output, for every input sample, a member of $\mathcal{H}$ that classifies correctly as many sample points as any member of $\mathcal{H}$ can classify *with margin* $> \mu$ (where the margin of a point relative to a hypothesis is the radius of the largest ball around the point that does not intersect the boundary of the hypothesis).

Approximation algorithms for the corresponding minimization problems in the cases of half-spaces, monomials and axis-aligned rectangles follow from the work of Kearns and Li [17] (see [15, 19]). An efficient algorithm for maximizing agreement with a sample using axis-aligned rectangles in the case of a constant number of attributes was described by Dobkin, Gunopulos and Maass [9].

ACKNOWLEDGMENTS

Phil Long acknowledges the support of National University of Singapore Academic Research Fund Grant RP960625.

References

[1] Edoardo Amaldi and Viggo Kann. The complexity and approximability of finding maximum feasible subsystems of linear relations. *Theoretical Computer Science*, 147:181–210, 1995.

[2] D. Angluin and P. D. Laird. Learning from noisy examples. *Machine Learning*, 2:343–370, 1988.

[3] Sanjeev Arora, Laszlo Babai, Jacques Stern, and Z. Sweedyk. Hardness of approximate optima in lattices, codes, and linear systems. *Journal of Computer and System Sciences*, 54(2):317–331, 1997.

[4] Peter Bartlett and Shai Ben-David. Hardness results for neural network approximation problems. *The 1999 IMA European conference on Computational Learning Theory*, pages 50–62, 1999.

[5] Shai Ben-David, Nadav Eiron, and Hans U. Simon. The computational complexity of densest regios detection. To appear in proc. of the 13th Conference on Computational Learning Theory, 2000.

[6] A.L. Blum and R.L. Rivest. Training a 3-node neural network is NP-complete. *Neural Networks*, 5(1):117–127, 1992.

[7] A. Blumer, A. Ehrenfeucht, D. Haussler, and M. K. Warmuth. Learnability and the Vapnik-Chervonenkis dimension. *JACM*, 36(4):929–965, 1989. Preliminary version in STOC'86.

[8] Bhaskar DasGupta, Hava T. Siegelmann, and Eduardo D. Sontag. On the complexity of training neural networks with continuous activation functions. *IEEE Transactions on Neural Networks*, 6(6):1490–1504, 1995.

[9] D. P. Dobkin, D. Gunopulos, and W. Maass. Computing the maximum bichromatic discrepancy, with applications in computer graphics and machine learning. *Journal of Computer and System Sciences*, 52(3):453–470, 1996.

[10] U. Feige. A threshold of $\log n$ for approximating set cover. *Proc. 28th Ann. ACM Symp. on Theory of Comp.*, pages 314–318, 1996.

[11] Y. Freund and R. Schapire. Experiments with a new boosting algorithm. In *Proc. 13th Int. Conf. on Machine Learning*, pages 148–156. Morgan Kaufman Publishers, 1996.

[12] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Proceedings of the Second European Conference on Computational Learning Theory*, pages 23–37, 1995.

[13] J. Håstad. Some optimal inapproximability results. *Proceedings of the 29th ACM Symposium on the Theory of Computing*, 1997.

[14] D. Haussler. Decision theoretic generalizations of the PAC model for neural net and other learning applications. *Information and Computation*, 100(1):78–150, 1992. Preliminary version in FOCS'89.

[15] D. P. Helmbold and P. M. Long. Tracking drifting concepts by minimizing disagreements. *Machine Learning*, 14(1):27–46, 1994.

[16] Klaus-U. Höffgen, Hans-U. Simon, and Kevin S. Van Horn. Robust trainability of single neurons. *J. of Comput. Syst. Sci.*, 50(1):114–125, 1995.

[17] M. Kearns and M. Li. Learning in the presence of malicious errors. *SIAM Journal on Computing*, 22(4):807–837, 1993.

[18] M. Kearns and L. Valiant. Cryptographic limitations on learning boolean formulae and finite automata. *Journal of the Association for Computing Machinery*, 41(1):67–95, 1994.

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

[20] M. Kharitonov. Cryptographic hardness of distribution specific learning. *Proceedings of the 25th ACM Symposium on the Theory of Computing*, pages 372–381, 1993.

[21] L. Pitt and L. G. Valiant. Computational limitations on learning from examples. *Journal of the Association for Computing Machinery*, 35(4):965–984, 1988.

[22] R. Schapire. The strength of weak learnability. *Machine Learning*, 5:197–227, 1990.