[6] Jaikumar Radhakrishnan. Better Bounds for Threshold Formulas, *Proc. 32nd FOCS*, pp. 314–323, 1991.

[7] P. Raghavan, Lecture Notes on Randomized Algorithms, Research Report, IBM Research Division, RC 15340 (#68237) 1/9/90.

[8] Joel Spencer, Ten Lectures on the Probabilistic Method, Monograph, *CBMS-NSF Regional Conference Series in Applied Mathematics*, 1987.

[9] L.G. Valiant, A Theory of the Learnable, *Communications of the A.C.M.,* 27(11), 1984, pp. 1134 – 1142.

[10] L.G. Valiant, Short Monotone Formulae for the Majority Function, *Journal of Algorithms*, **5**, 363 –366 (1984).

Our lower bound actually holds for a large class of boolean functions that includes majority. Let f be any boolean function and f_n be f restricted to n-bit inputs. If there is a constant c_f such that for large enough n, f_n is identically 0 on the bottom $c_f n$ levels of the lattice and identically 1 on the top $c_f n$ levels of the lattice, then any monotone formula computing f must have size at least $\Omega(n^2)$.

6 Conclusions and Open Problems

It would be interesting to find particular concept classes, such as boolean circuits, for which equivalence queries can be drawn from smaller hypothesis classes than the ones established here.

Our lower bound technique for majority can be described as a quick-and-dirty technique where the analysis is rather coarse. It is surprising that such a technique yields the same lower bound as more intricate combinatorial techniques. It would be interesting to explore this technique for other models such as non-monotone formulae, monotone circuits, etc. and for other functions such as the threshold functions with $o(n)$ thresholds. Obvious extensions don't seem to work.

7 Acknowledgements

Thanks are due to Mike Kearns who generally encouraged the bringing forth of this result and to Rob Schapire who through indirect communication raised the possibility of a connection with majority. Thanks also to Steven Rudich who pointed out that the lower bound holds for more than just the majority.

References

[1] D. Angluin, Queries and Concept Learning, *Machine Learning,* 2(4), 1988, pp. 319 – 342.

[2] V. Chvatal, Probabilistic methods in graph theory, *Annals of Operations Research* **1**, pp. 171–182 (1984).

[3] H. Chernoff, A measure of asymptotic efficiency for tests of a hypothesis based on sums of observations, *Annals of Math. Stat.*, **23** pp. 493 –509 (1952).

[4] V.M. Khrapchenko, A method of obtaining lower bounds for the complexity π-schemes, *Math. Notes Acad. Sci. USSR*, **11** pp. 474–479, (1972).

[5] M. Kearns, The Computational Complexity of Machine Learning, ACM Distinguished Disseration, 1989, MIT Press.

sum of the R_x's is equal to 0 when the expected sum is $2^{n/2}$ is given by $F^-(2^{n/2}, 1)$ which is double exponentially small.

Finally there are only single exponentially many choices for the $q(n)$ concepts to be chosen out of the 2^n concepts as inputs to m and there are only single exponentially many different possible circuits m. Thus the probability over all the choices of the $q(n)$ concepts and of the circuit m that the exact equivalence query for C_n is constructed by m is still almost 0. In any case it is nowhere near 1. Thus by the probabilistic method there is a concept class C_n for which membership and other equivalence queries don't work and the exact equivalence query cannot be constructed by any circuit with $\leq r(n)$ gates. This proves the lower bound.

5 Lower Bounds on Size of Monotone Majority Formulae

We have shown that for any concept class C_n where each point of the domain belongs to very few or most concepts, the majority of $O(n)$ concepts constructs the majority concept. On the other hand we have also shown that there is a concept class (where each point of the domain is in most or few concepts) such that no monotone formula of size $\leq (n/\log n)^2$ constructs the majority concept. In particular this implies a lower bound of $(n/\log n)^2$ for the size of any monotone formula for majority.

Actually we can sharpen this lower bound. When we are trying to prove a lower bound for majority, we do not have to worry about making membership and other equivalence queries ineffective. Consequently we define the concept class C_n by an experiment where each point in A belongs to each concept with probability $1/8$ and each point in B belongs to each concept with probability $7/8$. We then show that the following list of bad events have a very low cumulative probability.

1. Some point in A belongs to more than $1/4$ of the concepts.
2. Some point in B belongs to less than $3/4$ of the concepts.
3. A concept contains less than $1/16$ of the points in A
4. A concept contains more than $15/16$ of the points in B.

Events 1 and 2 are bad because they violate the promise under which the majority concept is constructed as a majority of $O(n)$ concepts and events 3 and 4 are bad because without them the input probabilities to the formula in the proof of lemma 1 are too small. Clearly once again, each event has a probability that is at most super exponentially small and once again a straight forward application of lemma 1 gives us a lower bound of $\Omega(n^2)$.

This is the best known lower bound for majority. It was proven first by Khrapchenko[4] who showed an $\Omega(nk)$ lower bound for the k threshold function. This was subsequently improved to $\Omega(nk\log(n/k-1))$ by Radhakrishnan[6] but this improvement does not produce an improvement in the lower bound for majority.

It is clear that the ordered pairs are a lower bound on the probabilities that they represent. Dependence has to be taken into account in computing these probabilities *if and only if* the $q(n)$ concepts that are input to the formula are not all distinct. But having two of the inputs be the same concept only helps us in proving the lower bound since it only slows down the rate of decrease of these probabilities. This statement relies on the fact that we are dealing with monotone formulae.

The following lemma tells us how fast these ordered pairs go down in value and is central to our lower bound proof.

Lemma 1 *If the inputs to a formula are all labeled with ordered pairs (p_1, q_1) and we require that the output have a label that is componentwise less than or equal to the ordered pair (p_2, q_2), then the formula must have $\frac{\log p_2 \log q_2}{\log p_1 \log q_1} - 1$ gates.*

Proof: We prove the statement by induction on the size of the formula. Assume wlog that the gate at the output is an AND gate. Let m_1 and m_2 be the two subformulae at the inputs to this AND gate. Let (p_a, q_a) and (p_b, q_b) be the labels of the outputs of m_1 and m_2 respectively. Then $(p_2, q_2) = (p_a p_b, q_a + q_b - q_a q_b)$.

First note that the base case, i.e. a formula with no gates satisfies the conditions of the lemma since in this case $(p_1, q_1) = (p_2, q_2)$ and the expression giving the lower bound is 0. Suppose that m_1 and m_2 satisfy the inductive hypothesis. We have

$$\begin{aligned} |m| &\geq |m_1| + |m_2| + 1 \\ &\geq \frac{\log p_a \log q_a + \log p_b \log q_b}{\log p_1 \log q_1} - 1 \\ &\geq \frac{\log p_2 \log q_2}{\log p_1 \log q_1} - 1 \end{aligned}$$

The last inequality above follows because $\log q_2$ is smaller in absolute value than $\log q_a$ and $\log q_b$ because $q_2 \geq \max(q_a, q_b)$.

A symmetric calculation works for OR gates as well. ∎

Corollary 1 *If the size of m is $\leq r(n)$ and the output wire is labeled with (p_2, q_2), then at least one of p_2 or q_2 is greater than $1/2^{n/2}$ for large enough n.*

Proof: The statement follows from a straight forward application of lemma 1 with $p_1 = q_1 = 1/(2n^{\sqrt{\alpha(n)}})$. ∎

By the corollary above m either has a 'high' probability of accepting a point in A or a high probability of rejecting a point in B. Suppose wlog that the former is true. Then each point in A is accepted with probability at least $1/2^{n/2}$. Since the cardinality of A is 2^{n-1} the expected number of points in A which are accepted by m is approximately $2^{n/2}$.

We will define a random variable R_x for each point $x \in A$. R_x is an indicator variable which is 1 if x is accepted by m and 0 otherwise. Note that the R_x's are non-negative, independent, identically distributed variables. By Chernoff bounds the probability that the

Suppose the learning algorithm makes n^k queries (where k is a constant) in order to learn a concept in C_n.

Choose n large enough that $\sqrt{\alpha(n)} > 2k+1$ and consider the concept class C_n defined above. A sketch of the argument proving the contradiction follows.

Essentially each point in A belongs to very few of the concepts and each point in B belongs to most of the concepts. Thus if the adversary always answers NO for any membership query from A and always answers YES for a membership query from B, then even after n^k membership queries most concepts in C_n have still not been eliminated. The same argument holds when an equivalence query is made which either includes at least one point in A or excludes at least one point in B. The adversary can always present the included/excluded point as a counter-example again ensuring that a very small fraction of the concepts get eliminated. Thus the only query that makes enough progress to allow the algorithm to finish in n^k queries is an equivalence query which is the majority concept query.

We now estimate the probabilities of our first two bad events. Let E_1 be the event that some point in A is in more than $2^n/n^{2k}$ concepts. Then by Chernoff bounds $\Pr(E_1) < 2^n F^+((2^n)n^{-\sqrt{\alpha(n)}}, n^{\sqrt{\alpha(n)}-2k}-1)$ which has a crude upper bound of $2^{-2^{n/2}}$. Let E_2 be the event that some point in B is absent from more than $2^n/n^{2k}$ concepts. The probability of E_2 has an identical bound and together these probabilities are super exponentially small.

We now have to show that m has an extremely small probability of arriving at the exact equivalence query.

We need to define the third and fourth bad events E_3 and E_4 at this point. Each concept in C_n is expected to have $2^{n-1}/n^{\sqrt{\alpha(n)}}$ points from A and $2^{n-1}(1-1/n^{\sqrt{\alpha(n)}})$ points from B. E_3 represents the event that some concept has too few points from A and E_4 the event that some concept has too many points from B. These events are bad because such a concept is too good an approximation to the majority concept. More precisely, let E_3 be the event that some concept in C_n has less than $2^{n-1}/2n^{\sqrt{\alpha(n)}}$ points from A. Then $\Pr(E_3) < 2^n F^-(2^{n-1}/n^{\sqrt{\alpha(n)}}, 1/2)$. Thus the probability of E_3 is upper bounded once again by a super-exponentially small value. Symmetrically we let E_4 represent the event that some concept in C_n is missing less than $2^{n-1}/2n^{\sqrt{\alpha(n)}}$ points from B and the upper bound on E_4 is identical to the upper bound on E_3.

In what follows we make the assumption that events E_1, E_2, E_3, E_4 do not happen. This assumption introduces dependencies in the rest of the analysis. However, since the E_i are such low probability events, the effect of the dependencies can be ignored.

Suppose we maintain ordered pairs (p_1, p_2) at each wire in the formula where p_1 represents the probability of that wire being 1 when the input is a random point in A and p_2 is the probability of that wire being a 0 when the input is a random point in B. At the $q(n)$ inputs to the formula the ordered pairs are lower bounded by $(1/2n^{\sqrt{\alpha(n)}}, 1/2n^{\sqrt{\alpha(n)}})$.

It is easy to compute (assuming independence) the ordered pairs in the rest of the formula. Suppose (p_1, p_2) and (q_1, q_2) are the ordered pairs at the input to a gate. If the gate is an AND gate the ordered pair at the output is $(p_1 q_1, p_2 + q_2 - p_2 q_2)$ and if the gate is an OR gate the ordered pair at the output is $(p_1 + q_1 - p_1 q_1, p_2 q_2)$.

a concept in polynomially many queries. Admittedly, the concept class C_n will be highly artificial and even non-evaluatable in the sense defined in [5]. However, it just illustrates that any generic argument about the number of queries required to learn a concept class must allow equivalence queries from a hypothesis class essentially as large as the one defined in the previous section. It is still possible that for specific concept classes such as boolean circuits of a bounded size, there is a polynomial sequence of queries that allows one to learn, where the equivalence queries are drawn from a smaller hypothesis class.

We introduce the following notation. We say that an equivalence query belongs to $H(C_n, p(n))$ if it can be expressed as a $p(n)$-sized formula. For the lower bound we will focus on the set $\mathcal{C}_n$ of concept classes C_n where $|C_n|$ is bounded by 2^n and X_n, the domain of C_n is $\{0,1\}^n$. For such classes the procedure of the previous section makes only polynomially many queries to learn a target concept. Suppose now that there is a *uniform* bound $r(n) \in o((n/\log n)^2)$ such that every concept class $C_n \in \mathcal{C}_n$ is learnable in polynomially many queries with equivalence queries drawn from $H(C_n, r(n))$. We know that $r(n) = (\frac{n}{\log n \alpha(n)})^2$ where $\alpha(n) \to \infty$ as $n \to \infty$.

Let m be any monotone formula which takes $q(n)$ inputs, has no more than $(r(n))$ gates and produces one output. For the lower bound we restrict consideration to formulae m with fan-in 2.

We define a distribution on concept classes out of which a concept class C_n is chosen as follows:

Each concept in the concept class C_n will be a subset of the set of n-bit strings. Let A denote the set of n-bit strings beginning with a 0 and B denote the set of n-bit strings beginning with a 1.

The class C_n will have 2^n concepts, $c_1, c_2, \ldots, c_{2^n}$. Each c_i is randomly defined in the following manner: Each $x \in A$ belongs to c_i with probability $n^{-\sqrt{\alpha(n)}}$ and each $y \in B$ belongs to c_i with probability $1 - n^{-\sqrt{\alpha(n)}}$.

These random choices are to be thought of as being made *after* the particular circuit m and the particular set of $q(n)$ concepts whose outputs are input to m have been chosen. We can then talk about the probability of m's output satisfying certain properties.

The overall argument will have the following form. In our probability space, we will show that membership queries and most equivalence queries are ineffectual with very high probability. There is only one particular equivalence query that can make sufficient progress to allow the algorithm to finish in polynomially many queries.

We will then show that as we consider all possible formulae of size $\leq r(n)$ and all possible subsets of $q(n)$ of the 2^n concepts in C_n as inputs to the formulae, the total probability that any of these formulae constructs the desired equivalence query is negligibly small. By the probabilistic method we know that there is a concept class C_n which cannot be learnt as long as equivalence queries are drawn from unions and intersections of $r(n)$ concepts.

In the course of this proof we will identify several 'bad events' which will just be events that make the concept class C_n not suitable for the lower bound proof. We will denote these events by E_i and show that the sum of their probabilities is strictly less than 1 thereby showing that there is some concept class C_n for which the lower bound holds.

Consider a depth 2 formula, m, where the gates at level 1 are ORs and the output gate is an AND. Suppose each OR gate has fan-in $2n/\log n$ and the AND gate has fan-in $2n/\log\log n$. This implies that m is of size $O(\frac{n^2}{\log n \log\log n})$ and has $\frac{4n^2}{\log n \log\log n}$ inputs. Suppose these inputs are randomly and independently chosen concepts from S. We will show that m has a non-zero probability of computing the majority concept.

Let x be an input that is accepted by at least $\frac{n-1}{n}$ of the concepts in S. The probability that any particular OR gate does not accept x is at most 2^{-2n} and the probability that there exists an OR gate which does not accept x is at most $(2n)2^{-2n}$ which is less than 2^{-n} for large enough n.

Now let y be an input that is accepted by at most $1/n$ of the concepts in S. The probability that a particular OR gate accepts y is at most $1-(1-\frac{1}{n})^{2n/\log n}$ which is at most $2/\log n$. The probability that m accepts y is at most $(2/\log n)^{2n/\log\log n}$ which is less than 2^{-n} for large enough n.

There are 2^n inputs, each of which 'behaves badly' with probability less than 2^{-n}. Thus there is a nonzero probability that all the inputs behave well on m. By the probabilistic method there exists an m for which all the inputs behave well, i.e such that m accepts exactly those inputs which are accepted by most concepts.

It is clear that the algorithm described above makes only polynomially many queries and that its equivalence queries are of size $\frac{n^2}{\log n \log\log n}$.

3.3 An Optimal Algorithm

We now sketch an algorithm whose equivalence queries are of size $O((n/\log n)^2)$. This will be shown to be optimal in the next section.

Once again we assume that using membership queries we have come to a stage where every point of the domain is in at most $1/n$ of the concepts in the viable concept set S, or in at least $1-1/n$ of these concepts.

The formula that computes the majority concept in optimal size is a complete binary tree of depth $2(\log n - \log\log n + 1)$ where the odd levels have OR gates and the even levels have AND gates. The proof that such a formula exists is again by the probabilistic method and the details of the analysis are not given here since they are somewhat similar to the lower bound analysis of the next section.

We note that all three algorithms described here can be implemented in PSPACE provided there is a PSPACE test of membership of a point in a concept.

4 A Lower Bound

In this section we show that the results of the previous section are optimal in a certain sense. Specifically, we show that there is a parametrized concept class, C_n, for which learning a target concept in polynomially many queries requires that equivalence queries be drawn from the set consisting of $\Omega((n/\log n)^2)$-sized formulae. In other words when equivalence queries are restricted to be drawn from a smaller hypothesis class, no learning algorithm can learn

Let $r_1, r_2, \ldots, r_{24n}$, be $24n$ concepts that are randomly and independently chosen from S. Consider an equivalence query of the form $f = \mathrm{Maj}(r_1, r_2, \ldots, r_{24n})$. We want to show that there is a non-zero probability that this hypothesis concept consists exactly of the points in the domain that belong to at least three quarters of the concepts in S. For x such that x belongs to at most one quarter of the concepts in S, the probability that $f(x) = 1$ is upper bounded by $F^+(6n, 1) = (e/4)^{6n} < 2^{-n}$. This follows from equation (1) above. Similarly the probability that a y that is accepted by most circuits in S is rejected by this new circuit is upper bounded by $F^-(18n, 1/3) = \exp(-n) < 2^{-n}$. The overall probability that some point of the domain is wrongly classified by this circuit is less than 1. Thus by the probabilistic method there exists a formula of the form described above that constructs the majority concept.

We now make an equivalence query using the majority concept. It is clear that any counter-example to this query eliminates at least $3/4$ of the concepts in S and hence at every stage we eliminate at least $1/4$ of the concepts in S. The bound of $O(\log |C|)$ on the number of queries follows.

The best upper bound known on the size of majority formulae on $O(n)$ inputs is due to Valiant[10] and is $O(n^{5.3})$. A lower bound of n^2 is known on this size[4].

3.2 A Depth 2 Formula for the Majority Concept on Very Biased Inputs

We modify our techniques of the last subsection to give a more elementary construction of the majority concept in the case where points of the domain either belong to overwhelmingly many or to very few concepts in S. Our construction here assumes that each input either belong to at most $1/n$ of the remaining concepts or to at least $1 - 1/n$ of the remaining concepts. This can be easily arranged using membership queries. Under this assumption we construct a depth 2 formula with $O(\frac{n^2}{\log n \log\log n})$ inputs which computes the majority concept. Combinatorially this construction is simpler than the majority circuit of the previous subsection. Also as a special application the technique of this subsection allows the learning of AC^0 circuits of size bounded by $p(n)$ by using AC^0 circuits of size $O(n^2 p(n))$ as equivalence queries. One final advantage of this construction is that it uses provably fewer unions and intersections than the majority-based construction in its equivalence queries.

Let $c \in C_n$ be the target concept. Roughly our bound is proved as follows. At any stage let $S \subseteq C_n$ be the set of concepts that are consistent with all queries so far. As long as there is a membership query whose result will reduce the size of S by a factor of $1/n$ or more, we make that membership query. Otherwise, we have a situation where each input is accepted by 'most' concepts in S or rejected by 'most' concepts in S. In this case we create a formula, m, which accepts exactly those inputs which are accepted by most concepts in S. Presenting m as an equivalence query will drastically reduce the size of S and we will certainly achieve the reduction in the size of S by a polynomial factor. Since C_n is only exponentially big, it follows that in polynomially many queries the size of S is reduced to 1, i.e. the target concept has been identified.

2.2 Chernoff bounds

We make extensive use of Chernoff bounds through out this paper. These bounds on tails of distributions of sums of independent, identically distributed, non-negative random variables were first proved by Chernoff[3]. A very readable discussion and proofs of these bounds can be found in the lecture notes on randomized algorithms by Raghavan[7]. To make this paper self contained we list the two inequalities that we will repeatedly use. The statements here are identical to those in [7].

Let $X_1, X_2, \ldots, X_n$ be independent Bernoulli trials with $Pr(X_i = 1) = p_i, p_i \in (0,1)$. Let $X = \sum_{i=1}^{n} X_i$ and $\mu = \sum_{i=1}^{n} p_i > 0$. Then

$$\Pr(X > (1+\delta)\mu) < \left[\frac{\exp(\delta)}{(1+\delta)^{(1+\delta)}}\right]^{\mu} = F^{+}(\mu, \delta). \tag{1}$$

Under the same hypothesis as above, for $\delta \in (0,1]$,

$$\Pr(X < (1-\delta)\mu) < \exp(-\mu\delta^2/2) = F^{-}(\mu, \delta). \tag{2}$$

Several generalizations of Chernoff bounds to not necessarily 0-1 random variables, to not necessarily identically distributed random variables etc. are known and widely used. For results such as these, see for example, Chvatal[2].

2.3 The Probabilistic Method

Roughly, the probabilistic method is used to prove the existence of combinatorial objects having certain properties. This is done by setting up a probability space whose points are combinatorial objects and showing that a random point chosen from it has a probability greater than 0 of having the desired property. For an excellent treatment of the probabilistic method see Spencer[8].

3 Learning Concept Classes

In this section we give three different learning algorithms for a learning concept class $C = \cup C_n$ satisfying the conditions of the previous section. All these algorithms make polynomially many queries but use equivalence queries of different sizes.

3.1 Algorithm using Majority

Let $c \in C_n$ be the target concept. Our algorithm maintains a set S of concepts that are consistent with all the queries so far. Initially $S = C_n$. If there is a membership query that eliminates at least a quarter of the concepts in S regardless of what the answer is, we make that membership query. Otherwise, for each $x \in \{0,1\}^n$ either x is in at least three quarters of the concepts in S or x is in at most one quarter of the concepts in S.

express equivalence queries in an efficient learning algorithm. Our main results are

1. If majorities of $O(n)$ concepts in C_n are permitted as equivalence queries then there are polynomially many queries that allow learning.

2. If in learning C_n, formulae involving $O((n/\log n)^2)$ unions and intersections of concepts in C_n are permitted as equivalence queries then there are polynomially many queries which allow exact identification of the target concept.

3. If equivalence queries are uniformly restricted to be from the class of formulae using $o((n/\log n)^2)$ unions and intersections of concepts in C_n then there is a class C for which learning is not possible using polynomially many queries.

4. Any monotone formula computing majority and certain other boolean functions of n inputs must have size at least $\Omega(n^2)$.

2 General Framework

2.1 The Model

We will restrict ourselves to concept classes, C, for which the number of concepts in C_n is upper bounded by $2^{p(n)}$ for some polynomial $p(n)$. We will also assume for simplicity that the domain of C_n is $X_n = \{0,1\}^n$. For such classes we study learning algorithms that make polynomially many queries.

In order for a learning algorithm to make only polynomially many queries, it must be allowed to make equivalence queries from a larger concept class called the *hypothesis class*. We are interested in investigating the 'complexity' of the hypothesis class.

We use the vehicle of monotone formulae to quantify this complexity. A monotone formula is a circuit consisting of AND and OR gates such that all the gates have fan-out 1. We will restrict ourselves to equivalence queries which can be expressed as unions (ORs) and intersections (ANDs) of concepts in the concept class being learnt. Any such equivalence query can be expressed as a monotone formula, f, whose inputs are concepts $c_1, c_2, \ldots$. The concept represented by f consists of all points $x \in X_n$ such that $f(x) = f(c_1(x), c_2(x), \ldots) = 1$.

We use the standard definition of the *size* of a formula. We will let the size be the number of wires in the formula. This means that we can transform any formula to have fan-in 2 without blowing up its size by more than a constant factor. When we restrict ourselves to formulae with fan-in 2, we will let the size of the formula be the number of gates.

We will measure the complexity of a learning algorithm by the size of its hypotheses when expressed as formulae over the concept class being learnt. We will prove tight bounds on this complexity in the next two sections.

1 Introduction

The notion of distribution-free machine learning of concepts was introduced by Valiant[9]. Concepts are subsets of a set called the domain. In the model introduced in [9] the learning algorithm is given positive and negative examples (points in the domain) of the concept to be learnt. These examples are assumed to be drawn from a fixed but unknown distribution and the goal of the learning algorithm is to come up with a hypothesis that 'closely' (according to the measure defined by the underlying distribution) matches the concept being learnt. Since the input to the algorithm is a random sample we must allow for error in the output, i.e. some probability that the output hypothesis will not closely match the target concept. We also cannot hope to achieve exact identification of the target concept.

Angluin[1] formalized an alternate model of learning based on queries to a teacher. In this model the learning algorithm is allowed to ask various questions about the target concept and these questions are answered (correctly) by the teacher. Several types of queries are considered in [1]. In this paper we will focus on two specific types of queries, *membership queries* and *equivalence queries*. Suppose the domain is X and the target concept is L. In a membership query the learning algorithm presents a point $x \in X$ and the teacher answers *yes* if $x \in L$ and *no* otherwise. In an equivalence query the learning algorithm presents a set $L' \subseteq X$ to the teacher and the teacher answers *yes* if $L = L'$. If not the teacher presents a *counter-example* i.e. a point x that lies in the symmetric difference of L and L'.

In both the models above one deals with concept classes rather than individual concepts so that one can talk about the asymptotic complexity of learning algorithms for particular concept classes. Many naturally occurring concept classes are parametrized by an integer parameter n. For instance the set of all boolean circuits of size bounded by a fixed polynomial in the number of inputs to the circuit represents a concept class and it is naturally parametrized by n, the number of inputs to the circuit. Thus this concept class C can be thought of as the union, $\cup C_n$ where C_n is the subset of C consisting of the n input circuits.

We are concerned with information-theoretic bounds on the number of (equivalence and membership) queries needed to exactly learn such classes. We have to decide what kinds of equivalence queries are permissible before we can prove these bounds. Angluin[1] points out that if the concepts that are queried as equivalence queries are restricted to be from the class being learnt, then there are concept classes for which each query eliminates only one concept from consideration and hence exponentially many queries are required.

Suppose S is any set of concepts over the domain, X. We can define a *majority concept*, m, for S by letting m consist of all $x \in X$ which are included in at least $1/2$ the concepts in S. Angluin[1] points out that if equivalence queries are allowed to query about majority concepts of arbitrary subsets of concepts in C_n, then learning can be achieved in $O(\log |C_n|)$ queries. This is the take-off point for this paper.

We first show that after suitable use of membership queries, the majority concept is not much more 'complex' than the concepts in the concept class being learnt. We then treat unions and intersections as the basic operations on concepts and look at bounds on the number of unions and intersections of concepts (in the class being learnt) that are needed to

On the Query Complexity of Learning and a Technique for Lower Bounds on Monotone Formulae

Sampath K. Kannan[1]

TR 91-33

Abstract

We consider the problem of learning parametrized concept classes with membership and equivalence queries. If C_n is the concept class being learned, we show that if equivalence queries can be made from a larger but still 'reasonable' hypothesis class, then there exist $O(n \log |C_n|)$ queries that exactly learn the target concept $c \in C_n$. We also show that our results are best possible in terms of how big the hypothesis class needs to be and thereby give a way of deriving a lower bound of $\Omega(n^2)$ on the size of monotone formulae for majority and other boolean functions. This matches the best known lower bounds for majority.

Department of Computer Science
The University of Arizona
Tucson, AZ 85721

[1]Supported by NSF grant CCR 91-08969