COMS 4252: Introduction to Computational Learning Theory.
Author: Hila Becker
Date: December 15, 2004

Limits on Exact Learning from Membership and Equivalence Queries

Abstract

In this paper we look at different combinatorial properties of concept classes that give bounds on the query complexity of exact learning. We consider learning from equivalence queries, membership queries and a combination of equivalence and membership queries. We examine and present proofs regarding efficient query-learnability as it relates to the notions of polynomial certificates, approximate fingerprints and the teaching dimension.

1 Introduction

In our study of learning theory, we have primarily focused on a learning model known as Probably Approximately Correct (PAC), where a learner is able to identify a close approximation of the target concept. While the PAC model enables efficient learning of concept classes, including decision lists and boolean conjunctions, it provides less satisfying results for learning more complex classes such finite automata and boolean formulas. We would therefore like to examine other learning models that produce more efficient results for learning such concept classes.

In this paper we will discuss and present results concerning a particular learning model known as Exact Learning, in which a learner obtains examples by making queries to an oracle. Developed primarily by Angluin, this model requires that the learner outputs a hypothesis that is exactly equivalent to the target concept. Exact learning with queries gives a learner the power to actively request examples from an oracle, in a fashion that will benefit its learning process thus improving the complexity of the learning. We are particularly be concerned with exact learning using two types of queries known as equivalence queries and membership queries. We consider three different learning settings using (i)equivalence queries (ii) membership queries and (iii) a combination of equivalence and membership queries. For each setting, we discuss properties that give limits on the number of queries needed to efficiently exactly learn a concept.

In particular, it was shown by Angluin that a concept class is not efficiently exactly learnable using equivalence queries alone if we can find a concept from our set of candidate hypotheses that is consistent with only a superpolynomially small number of hypotheses from the set. Such a concept is defined as an *approximate fingerprint* and we present the proof by [2] showing that a concept class that possesses the approximate fingerprints property cannot be efficiently exactly learnable from equivalence queries.

We also present a bound on the number of membership queries needed to learn a concept in terms of the teaching dimension. The teaching dimension from Goldman and Kearns [6] provides a measure of the smallest number of examples needed in order exactly identify a concept c. We can therefore present a proof showing that the teaching dimension gives a bound on the number of membership queries needed to efficiently exactly learn c.

For learning with equivalence and membership queries, we present very interesting results from [1] which relate efficient exact learning using equivalence and membership queries to a property of representation classes defined as polynomial certificates. We use this notion to show that a target class is polynomial query learnable with equivalence and membership queries if and only if it has polynomial certificates.

More specifically, we begin by discussing the polynomial query learnability of Boolean concept classes using equivalence and membership queries. We will define the idea behind polynomial certificates and show that a representation of a Boolean concept class is polynomial query learnable if and only if it has polynomial certificates [1]. Next we discuss Angluin's results in [2] concerning, proving that a concept class is not efficiently exactly learnable using equivalence queries if it has approximate fingerprints. We also show the results from Goldman and Kearns [6], proving a bound on the number of membership queries needed to learn in terms of the teaching dimension. Finally, we show that projection-closed Boolean concept classes are efficiently exactly learnable from membership queries only if they have a polynomial teaching dimension [1].

2 Preliminaries

A representation class $\mathcal{R}$ is defined by $\langle \Sigma, \Delta, R, \mu \rangle$ where Σ and Δ are finite alphabets, R is a subset of Δ^* and μ maps R to subsets of Σ^*. A concept c is a subset of Σ^* and R is the set of concept representations. μ is a mapping that determines whether a concept $c \in \Sigma^*$ is represented by a given representation from R. The concept class $\mathcal{C}$ is defined by the representation class $\mathcal{R}$ as the set of concepts that have representations in R. In this paper, we refer to the size of a concept c in terms of the smallest representation $r \in R$ such that $\mu(r) = c$. $\mathcal{C}_{m,n}$ is the concept class over n variables of length at most m. A Boolean concept class $\mathcal{C}_n$ is a concept class over n Boolean variables. an assignment to these variables is a string $x \in \{0,1\}^n$. For any $c \in C$, we use $c(x)$ to denote the classification of x by c, where $c(x) = 1$ if x is in c and $c(x) = 0$ otherwise.

In the model of exact learning, we are interested in finding a learning algorithm $\mathcal{A}$ such that for all $x \in X, X \subseteq \Sigma^*$ outputs a hypothesis h such that $h(x) = c(x)$. Therefore $\mathcal{A}$ outputs a representation of a concept that is exactly equivalent to the target concept. The learning algorithm can acquire the information to compute h by making queries to an oracle. The two kinds of queries that we discuss are equivalence queries and membership queries.

Definition 1 *An* Equivalence Query $EQ(h)$ *asks "is $h(x) = c(x)$ for all x?". The response to this type of query is either "yes" if the learner successfully identified the target concept, or a counterexample x where $h(x) \neq c(x)$.*

Definition 2 *A* Membership Query $MQ(x)$ *asks "is x a member of the unknown concept c?". The learner receives in response the value of c(x).*

The *query complexity* of $\mathcal{A}$ is the sum of the lengths of inputs to equivalence queries, membership queries and the counterexamples received in response to equivalence queries. Note that if $\mathcal{A}$ only uses membership queries then its query complexity is the sum of the length of inputs to membership queries, and if it only uses equivalence queries then its query complexity is the sum of the length of inputs to equivalence queries and counterexamples received as output.

A representation class $\mathcal{C}$ is *efficiently exactly learnable* using equivalence and membership queries if there exists a learning algorithm $\mathcal{A}$ and a two-variable polynomial p such that for any unknown $c \in C$ and any positive integer m:

- $\mathcal{A}$ has query complexity of at most p(m,n) at any stage of its execution, where m is the size of c and n is the maximum length of any counterexample received from an equivalence query.
- $\mathcal{A}$ outputs a hypothesis $h \in C$ such that $h(x) = c(x)$ for all instances x
- We refer to the "running time" of $\mathcal{A}$ as the bound on the number of queries made by $\mathcal{A}$ *and* the length of inputs to each of these queries, which is also p(m,n).

This definition is slightly modified when we discuss learning from equivalence or membership queries alone.

3 Learning with Equivalence and Membership Queries

In this section, we discuss the results given by [1] regarding the learnability of Boolean concept classes from membership and equivalence queries. We show that polynomial query learnability of a representation class $\mathcal{R}$ is implied by a characteristic known as polynomial certificates.
Let $\mathcal{C}$ be a Boolean concept class defined by the representation class $\mathcal{R}$. We present the *halving algorithm* for learning $\mathcal{C}$, which outputs hypotheses according to a majority vote. We would like to show the conditions under which it is possible to efficiently exactly learn a target concept class using this algorithm. The input to this algorithm is n, the number of variables in the target concept $c \in \mathcal{C}$. Let $H = \mathcal{C}_n$ be the initial set of all potential hypotheses. At each stage of the algorithm, H contains the concepts that are consistent with the labeled examples known to H thus far. For any set T of concepts, the *majority concept* denoted $MAJ(T)$ is a concept which contains an instance x if and only if at least half of the concepts in T contain x. The algorithm repeats the following steps until it receives a 'Yes' from an equivalence query

- Find the majority concept h of the set H
- Make an equivalence query $EQ(h)$
- If a counterexample $\langle x, c(x) \rangle$ was received, eliminate all concepts $h' \in H$ where $h'(x) \neq c(x)$. Otherwise, return h.

The number of queries used by the halving algorithm is $\lceil log_2 |\mathcal{C}_n| \rceil$ since every iteration eliminates at least half of the candidate hypotheses in H. In order to show that $\mathcal{C}$ is polynomial query learnable using the halving algorithm we must overcome two problems. We first argue that the query complexity of $\mathcal{C}$ is $poly(n, size(c))$. If we know the value of $m = size(c)$ we can initially set $H = \mathcal{C}_{m,n}$ so the algorithm makes at most $\lceil log_2 |\mathcal{C}_{m,n}| \rceil$ queries which is polynomial in m. If we do not know the size of c, we can run the algorithm with $H = \mathcal{C}_{l,n}$ with $l = 1$. If H is empty and no equivalence query returned a 'Yes', then we double l and run the algorithm again. Since in the worst case we will run the algorithm $log(m)$ times, the number of queries used is at most $\lceil log_2 |\mathcal{C}_{2m,n}| \rceil (log_2 m)$ which is also $poly(m, n)$.

According to our definition of polynomial query learnability, we must also show that the hypothesis h which we use to call $EQ(h)$ has a polynomial size representation in $\mathcal{R}$, or give an alternate way to achieve a polynomial bound on the query complexity. To show that polynomial query complexity is achievable, we will first need to define a property called Polynomial Certificates. Polynomial Certificates were first introduced by Hellerstein et al. [1].

3.1 Polynomial Certificates and Polynomial Query Learnability

Consider a Boolean concept class $\mathcal{C}$ and its representation class $\mathcal{R}$. We say that $\mathcal{R}$ has *polynomial certificates* if there exist two-variable polynomials p and q such that for all $m, n > 0$ and all concepts $h \in \{0,1\}^n$ with $size(h) > p(m,n)$, there is a set $Q \in \{0,1\}^n$ where $|Q| \leq q(m,n)$ and for all concepts $c \in \mathcal{C}$ there exists at least one $x \in Q$ such that $c(x) \neq h(x)$.
We will proceed to present the important result by [1] that gives a condition on polynomial query learnability from equivalence and membership queries.

Theorem 3 *A representation class $\mathcal{R}$ is polynomial query learnable with equivalence and membership queries if and only if $\mathcal{R}$ has polynomial certificates.*

Proof We will first show that if $\mathcal{R}$ has polynomial certificates then it is polynomial query learnable. This is achieved using a modification of the halving algorithm.
Assume we want to learn a target concept $c \in \mathcal{C}$ where $\mathcal{C}$ has a representation class $\mathcal{R}$ with polynomial certificates. Since $\mathcal{R}$ has polynomial certificates, there exist polynomials $p(m,n)$ and $q(m,n)$ where n is the number of variables in c and m is the size of c. The learning algorithm works as follows:

- Initialize $H = \mathcal{C}_{m,n}$
- Repeat the following steps until an equivalence query returns Yes
  - (i) If the size of $MAJ(H) \leq p(m,n)$ then make a query $EQ(h)$ where $h \in R$ is a representation of MAJ(H) and $|h| \leq p(m,n)$. If we receive a counterexample $\langle x, c(x) \rangle$ eliminate all concepts $h' \in H$ where $h'(x) \neq c(x)$.
  - (ii) If the size of $MAJ(H) > p(m,n)$ then we know that since $\mathcal{R}$ has polynomial certificates we can find a set Q such that $|Q| < q(m,n)$ and for all concepts in $c' \in \mathcal{C}$ there is some $x \in Q$ where $c'(x)$ is different than the labeling of x by MAJ(H). For all $x \in Q$ make a query MQ(x) and receive the value c(x). For some $x \in Q$, eliminate all concepts $h' \in H$ where $h'(x) \neq c(x)$.

We would like to show that this algorithm only make $poly(m,n)$ number of queries. At each iteration, we cut the number of concept in H by at least one half since both (i) and (ii) can identify a counterexample to the majority concept and eliminate the concepts in H that this counterexample is not consistent with. In (i) it is easy to see that the algorithm gets the counterexample from the equivalence query. In (ii) the algorithm can find x using the fact that $\mathcal{R}$ has polynomial certificates and therefore there must exist some $x \in Q$ that the target concept and the majority concept label differently. Since there are less than $q(m,n)$ elements in Q, the number of queries made by step (ii) is polynomial in m,n. The total number of queries made by this algorithm is therefore $\lceil log_2 |\mathcal{C}_{m,n}| \rceil$, which is polynomial in m and n and so the total query complexity of this algorithm is also polynomial in m,n.

We now show that if $\mathcal{R}$ is polynomial query learnable then it must have polynomial certificates. Consider an algorithm L that learns $\mathcal{R}$ with polynomial query complexity. Assume that the query complexity of L is p(m',n') for some polynomial p, where m' is the size of the target concept c' and n' is the number of variables in c'. Let c be a Boolean concept over n variables and $size(c) > p(m,n)$ for an arbitrary m. We construct the learning algorithm L such that it terminates if either (a) it's

query complexity exceeds p(m,n) or (b) L receives a 'Yes' in response to a call to $EQ(h)$ and outputs the correct hypothesis h. L has access to a membership query oracle $MQ(x)$ for $x \in \{0,1\}^n$, which returns the value c(x). L also has access to an equivalence query oracle $EQ(h)$ which will return a counterexample x where $h(x) \neq c(x)$. Note that if $|h| > p(m,n)$ the algorithm will terminate since the query complexity will exceed p(m,n). Therefore any call to $EQ(h)$ guarantees that $|h| \leq p(m,n)$ and so $h \neq c$ and we can always find a counterexample x.
Let Q contain $x \in \{0,1\}^n$ if x is either a counterexample returned to L from an equivalence query or an input to a membership query made by L. By our assumption regarding the query complexity of Q we can easily see that $|Q| \leq p(m,n)$. To complete the proof we must only show that there is no concept in the class $\mathcal{C}_{m,n}$ that is consistent with c on all $x \in Q$.
By contradiction, assume that there exists a concept d such that $d(x) = c(x)$ for all $x \in Q$. This means that d agrees with our target concept on all labeling of examples returned from membership queries and all counterexamples returned from equivalence queries. Since we assumed that L terminates when it receives a 'Yes' from a membership query where L's query complexity at most $p(m,n)$, we know that L must ask an equivalence query $EQ(r)$ where $|r| \leq p(m,n)$ is a hypothesis representation of d. By our definition of EQ, L would receive a counterexample x such that $d(x) \neq c(x)$. Since x is a counterexample returned to L by an equivalence query, it is a member of the set Q. This contradicts our assumption about d, since now there is an element in Q where $d(x) \neq c(x)$. ■

This theorem also holds for representation classes of general concept classes. The definition of polynomial certificates and the proof of the theorem can be extended to apply to classes with concepts defined over Σ^* as opposed to $\{0,1\}^n$. The main difference stem from the fact that in learning general concepts classes we must a way to show that the length of any counterexample is also polynomial in m and n. However, these result will not be discussed in this paper.

4 Learning From Equivalence Queries

We would like to explore limitations on exact learning of concept classes using equivalence queries alone. We will present the results of [2] which showed that certain concept classes such as DFA and DNF are not exactly learnable using equivalence queries. The proof of these results relies on the existence of a property of a concept's representation class, known as *approximate fingerprints*.

The idea behind approximate fingerprint is that they allow an adversary to find a counterexample that eliminates only very few concepts from the learner's set of candidate hypotheses. The representation classes that have approximate fingerprints share a similar structural characteristic; every concept in the target concept class $\mathcal{C}$ has a polynomial size representation but the majority concept $MAJ(\mathcal{C})$ does not.

Definition 4 (From [2]) ***Approximate Fingerprints.*** *Consider a concept class T,* $x \in \Sigma^*$*,* $\mathrm{b} \in \{0,1\}$ *and* $\alpha \in \mathbb{R}$ *such that* $\alpha > 0$*.* $\langle x, \mathrm{b} \rangle$ *is an* α*-approximate fingerprint with respect to T if*

$$|\{c \in T : c(x) = \mathrm{b}\}| < \alpha |T|$$

This simply states that the number of concepts $c \in T$ such that $c(x) = b$ is less than an α-fraction of the number of concepts in T. In order to show the conditions under which a representation class has approximate fingerprints, we need look at a *sequence of concept classes* $S = C_1, C_2, C_3, ...$ where each C_n represents a concept class. A *sequence of concept classes* is said to be bounded by f(n) with respect to $\mathcal{R}$ if every concept in C_n has a representation $r \in R$ whose length is at most $f(n)$ for a large n.
A representation class $\mathcal{R}$ has *approximate fingerprints* if an only if there exist two positive nondecreasing polynomials $p_1(n)$ and $p_2(n)$ such that for every positive nondecreasing polynomial q(n) we can find a sequence of concept classes $T_1, T_2, T_3, ...$ bounded by $p_1(n)$ with respect to $\mathcal{R}$ where $|T_n| \geq 2$ and if $r \in R, |r| \leq q(n)$ then there is $x \in \Sigma^*, |x| \leq p_2(n)$ and for $\mu(r) = c'$

$$|\{c \in T_n : c(x) = c'(x)\}| < |T_n|/q(n)$$

This means that $\langle x, c'(x)\rangle$ is an α-approximate fingerprint with respect to T_n where $\alpha = 1/q(n)$.

Theorem 5 *If a representation class $\mathcal{R}$ has approximate fingerprints then $\mathcal{R}$ is not efficiently exactly learnable using only equivalence queries*

The proof of this theorem relies on the intuition discussed earlier, where an adversary can return a counterexample x in response to an equivalence query made by the learner such that x is an approximate fingerprint of the class of potential hypotheses known to the learner. Thus, the learner can only eliminate a superpolynomially small number of hypothesis with every equivalence query made. This implies that the learner cannot identify the target concept in polynomial time. To make this more formal, consider the following proof by contradiction given by Angluin [2].
Proof Consider an algorithm $\mathcal{A}$ that can exactly learn $\mathcal{R}$ using equivalence queries and has a running time bounded by p(m,l) where m is the size of the smallest representation of a the target concept and l is the length of the longest counterexample received. Since $\mathcal{R}$ has approximate fingerprint we may assume the existence of polynomials $p_1(n)$ and $p_2(n)$ and let $q(n) = 2p(p_1(n)p_2(n))$. We also know that there exists a sequence of concept classes bounded by $p_1(n)$ with respect to $\mathcal{R}$ where $|T_n| \geq 2$ and if $r \in R, |r| \leq q(n)$ then there is $x \in \Sigma^*, |x| \leq p_2(n)$ and for $\mu(r) = c', \langle x, c'(x)\rangle$ is an α-approximate fingerprint with respect to T_n for $\alpha = 1/q(n)$.
Suppose that there exists an adversary that answers equivalence queries $EQ(r)$ made by our algorithm $\mathcal{A}$ for $r \in R$ using the following method:

- If $|r| > q(n)$ answer 'Yes'
- If $|r| \leq q(n)$ answer 'No' and return a counterexample x such that $x \in \Sigma^*, |x| \leq p_2(n)$ and for $\mu(r) = c', \langle x, c'(x)\rangle$ is a 1/q(n)-approximate fingerprint with respect to T_n.

Note that in order to exactly learn $\mathcal{R}$, our algorithm $\mathcal{A}$ needs to exactly learn each concept class T_n.

Claim 6 *For all $1 \leq i \leq q(n)/2$, $\mathcal{A}$ makes at least i equivalence queries and as a result, the number of inconsistent concepts that $\mathcal{A}$ can eliminate from the original $|T_n|$ known concepts is less than $(i/q(n))|T_n|$, and the length of any counterexample given is $\leq p_2(n)$.*

We proceed to prove this claim by induction.
Recall that our algorithm $\mathcal{A}$ runs in time $p(m, l)$ where m =size(c) which we defined to be the length

of c's representation $r \in R$. Since we know that the length of every representation in T_n is at most $p_1(n)$, $\mathcal{A}$'s running time is bounded by $p(p_1(n), l)$ with l being the length of the longest counterexample received or 0 if no counterexamples were received. In the case that $\mathcal{A}$ asks only one equivalence query $EQ(r)$, we can see that $|r| \leq p(p_1(n), 0) \leq q(n)$. Our adversary will therefore answer with a 'No' and return the counterexample x of length at most $p_2(n)$ where for $\mu(r) = c', \langle x, c'(x) \rangle$ is a $1/q(n)$-approximate fingerprint with respect to T_n. Therefore we have shown that when $\mathcal{A}$ makes one query it can eliminate less than $(1/q(n))|T_n|$ from the original known $|T_n|$ concepts.

Assume the claim is true for i such that $1 \leq i \leq q(n)/2$, we need to show that it holds for $i+1$. After the *ith* query, we know that by our assumption we can eliminate less than $(1/2)|T_n|$ concepts that are inconsistent with T_n on the counterexamples received by $\mathcal{A}$. Since $|T_n| \geq 2$ we can find at least two distinct concepts in T_n that agree on the labeling of all instances known to $\mathcal{A}$. $\mathcal{A}$ must therefore make another equivalence query EQ(r). Since the length of every counterexample x is at most $p_2(n)$, we can see that $|r| \leq p(p_1(n), p_2(n)) \leq q(n)$ and therefore the adversary will reply with 'No' and a counterexample x of length at most $p_2(n)$ where for $\mu(r) = c', \langle x, c'(x) \rangle$ is a $1/q(n)$-approximate fingerprint with respect to T_n. This counterexample eliminated less than $(1/q(n))|T_n|$ concepts in addition to the $(i/q(n))|T_n|$ concepts that were already eliminated. Therefore, after $i+1$ queries we have eliminated less than $((i+1)/q(n))|T_n|$ concepts, which by induction proves the claim.

To complete our proof we must show that no algorithm $\mathcal{A}$ can exist for exactly learning $\mathcal{R}$. After $\mathcal{A}$ makes $q(n)/2$ queries, we have shown that it eliminates less than half of the concepts in T_n that are inconsistent with the target concept. Since $|T_n| \geq 2$ we can find at least two distinct concepts in T_n that agree on the labeling of all instances known to $\mathcal{A}$. Since $\mathcal{A}$ did not receive a 'Yes' as a reply to an equivalence query it must request another EQ(r) from the adversary. However, since $\mathcal{A}$ already made $q(n)/2$ queries, its query complexity is $p(p_1(n), p_2(n))$ and therefore if $\mathcal{A}$ makes another query it will exceed its query complexity, which contradicts our assumption that $\mathcal{R}$ is efficiently-exactly learnable using $\mathcal{A}$. ■

5 Learning from Membership Queries

We would like to explore the bound on exact learning using only membership queries as it relates to a combinatorial property of known as the teaching dimension. The *Teaching Dimension* [6] of a concept class $\mathcal{C}$ is defined as the smallest number of labeled examples $\langle x, c(x) \rangle$ needed in order to exactly learn any concept $c \in \mathcal{C}$. We can think of it as an optimal mistake bound for exact-learning c.

A *teaching sequence* T for $c \in \mathcal{C}$ is a sequence of labeled examples such that c is the only concept in $\mathcal{C}$ that is consistent with T. $T(c)$ denotes all teaching sequences for c. More formally, the teaching dimension of a concept class $\mathcal{C}$ denoted $TD(\mathcal{C})$ is

$$TD(\mathcal{C}) = max_{c \in \mathcal{C}}(min_{\tau \in T(c)}|\tau|)$$

We will present two results relating the exact learnability of concept classes from membership queries to the teaching dimension. In [6] it was shown that any concept class is efficiently exactly learnable if it has a polynomial teaching dimension. The converse was proven in [1] for a special category of concept classes known as projection closed Boolean concept classes.

Theorem 7 *(from [6]) The number of membership queries made by a learner that exactly-learns a concept $c \in \mathcal{C}$ is at least $TD(\mathcal{C})$*

Proof In order to exactly-learn a concept $c \in \mathcal{C}$ using membership queries, a learner must return an exact representation of c given a set of labeled examples $\langle x, c(x) \rangle$. The learner can exactly identify c when it has enough examples to distinguish c from any other concept in $\mathcal{C}$. This is the set of examples that agree with c and not with any other concept in $\mathcal{C}$. The smallest such set for any $c \in \mathcal{C}$ is by definition TD(C). Therefore, the learner needs at least TD(C) examples in order to exactly learn c. ■

Consider a Boolean concept class $\mathcal{C}$ with a representation class $\mathcal{R}$ such that $\mathcal{C}_{m,n}$ is the class of Boolean concepts on n variables with a representation $r \in \mathcal{R}, |r| \leq m$ for $m, n > 0$. An example x consists of an assignment $x_i \in \{0, 1\}$ for $i = 1, 2, \ldots, n$. For $b \in \{0, 1\}$ define $x_{i \leftarrow b}$ as the assignment with x with the *ith* variable set to b. Let $c \in \mathcal{C}_{m,n}, b \in \{0, 1\}$ we say that c' is the concept $c_{i \leftarrow b}$ which is the result of projecting i to b in c, and $c'(x) = c(x_{i \leftarrow b})$.

Definition 8 *A concept class $\mathcal{C}$ is* projection-closed *if for $m, n > 0, c \in \mathcal{C}_{m,n} and b \in \{0, 1\}$, the size of $c_{i \leftarrow b}$ is at most m.*

We say that $\mathcal{C}$ is polynomial query learnable using membership queries if there exists an algorithm $\mathcal{A}$ and a polynomial $p(\cdot, \cdot)$ such that for all $m, n \geq 1$ and all concepts $c \in \mathcal{C}_{m,n}$, $\mathcal{A}$ can exactly identify c given n using at most $p(m, n)$ membership queries.

Theorem 9 *(from [1]) For a projection-closed Boolean concept class $\mathcal{C}$ with a representation class $\mathcal{R}$, if $\mathcal{C}$ has polynomial teaching dimension then $\mathcal{R}$ is polynomial query learnable from membership queries.*

The intuition behind the proof of this theorem is to first show that if $\mathcal{C}$ has a polynomial teaching dimension then it also has polynomial certificates. We have shown in a previous section that if $\mathcal{C}$ has polynomial certificates then it is polynomial query learnable using equivalence and membership queries. We can then easily show how to transform an algorithm that learns $\mathcal{C}$ using equivalence and membership queries into one that uses membership queries only and a total number of queries that is polynomial in the relevant parameters. This will complete the proof, showing that there exists an algorithm that exactly learns c using a polynomial number of membership queries.

Proof Let a two variable polynomial $p(\cdot, \cdot)$ be the teaching dimension of the projection-closed Boolean concept class $\mathcal{C}$. Consider a concept $c \subseteq \{0, 1\}^n$ which has a size greater than m. If $\mathcal{C}_{m,n}$ has only one concept d then it suffices to find one x such that $d(x) \neq c(x)$ since setting $Q = \{x\}$ proves the existence of a polynomial certificate. If $\mathcal{C}_{m,n}$ has more than one element then by definition of projection-closed Boolean classes we can show that it contains concepts corresponding to the 'True' and 'False' functions and therefore we can claim that there is an i, $0 \leq i < n$ and an example x with assignment $b_j \in \{0, 1\}$ to each x_j, $1 \leq j < n$ such that $c_{x_1 \leftarrow b_1, x_2 \leftarrow b_2, \ldots x_i \leftarrow b_i}$ is not in $\mathcal{C}_{m,n}$ but $c_1 = c_{x_1 \leftarrow b_1, x_2 \leftarrow b_2, \ldots x_i \leftarrow b_i, x_{i+1} \leftarrow 0}$ and $c_2 = c_{x_1 \leftarrow b_1, x_2 \leftarrow b_2, \ldots x_i \leftarrow b_i, x_{i+1} \leftarrow 1}$ are in $\mathcal{C}_{m,n}$. Let X_1, X_2 be the specification sets of c_1 and c_2, each of size at most $p(m, n)$. Define

$X'_1 = \{c_{x_1 \leftarrow b_1, x_2 \leftarrow b_2, \ldots x_i \leftarrow b_i, x_{i+1} \leftarrow 0} | x \in X_1\}$ and $X'_2 = \{c_{x_1 \leftarrow b_1, x_2 \leftarrow b_2, \ldots x_i \leftarrow b_i, x_{i+1} \leftarrow 1} | x \in X_2\}$. Since $c_{x_1 \leftarrow b_1, x_2 \leftarrow b_2, \ldots x_i \leftarrow b_i}$ is not in $\mathcal{C}_{m,n}$, no concept in $\mathcal{C}_{m,n}$ is consistent with c over $X = X'_1 \cup X'_2$ and since $|X| \leq 2p(m,n)$ we can say that $\mathcal{R}$ has polynomial certificates.

We must now show that we can translate the algorithm that learns a concept in $c \in \mathcal{C}_{m,n}$ from membership and equivalence queries into an algorithm that learns c using only membership queries. We show this by replacing every equivalence query by a polynomial number of membership queries. The existence of polynomial certificates implies the existence of an algorithm $\mathcal{A}$ that learns $\mathcal{C}$ using equivalence and membership queries with query complexity q(m,n) for some polynomial q. Since we assume that n is known, for $h \in \mathcal{C}$, the size of h is at most q(m,n) and therefore we can find a specification set S for h of size $p(q(m,n), n)$. If $\mathcal{A}$ makes an equivalence query with h we can simply make membership queries MQ(y) where $y \in S$ for each y, get the correct value of c(y) and then check if $c(y) = h(y)$. If this is true for every $y \in S$ return 'Yes' otherwise, pick an instance $y \in S$ where $c(y) \neq h(y)$ and return y as the counterexample. This yields an algorithm that exactly learns $\mathcal{C}$ using a polynomial number p(m,n) of membership queries. ■

We would also like mention briefly that similar results were presented by Hegedüs [4] who gave a bound on the number of membership queries needed for learning a concept class $\mathcal{C}$ in terms of a property he defined as the *extended teaching dimension*. The extended teaching dimension of $\mathcal{C}$ is the minimum number of elements in a specifying set S, which is a set defined for $h \in X$ such that there is at most one $c \in \mathcal{C}$ that is consistent with h on S. The exact bounds can be found in [5] to be similar to minimum value of q(m,n) in [1] defined as the bound on the size of the set Q.

6 Conclusion

In this paper we have shown the existence of combinatorial properties of concept classes that give bounds on the number of queries needed to exactly identify a concept. Although we have not shown specific applications of the results we discussed, such applications are the subject of many papers concerning exact learning from queries. More specifically, [2] presents negative results, showing that DFAs, NFAs, CFGs, DNF and CNF are not polynomially query learnable using only equivalence queries. On a positive note, it was shown [1,5] that DFA and monotone DNF are polynomial query learnable with equivalence and membership queries combined, due to the existence of polynomial certificates. [6] gives tighter bound in terms of the teaching dimension for classes such k-term DNF formulas and monotone decision lists. These are very exciting results since they prove that classes such as finite automata and boolean formulae that are not efficiently learnable in the PAC model, can be learned efficiently in the model of exact learning.

There are several papers that extend the results covered in this report. [8] Introduces a property called the consistency dimension which provides combinatorial bounds on learning with equivalence queries. The consistency dimension is then used in [7] along with polynomial certificates and the extended teaching dimension to give a new combinatorial characteristic on learning using membership queries, equivalence queries and a combination of membership and equivalence queries. This new parameter is claimed to be the exact-learning equivalent of the PAC model's VC dimension.

References

[1] Lisa Hellerstein, Krishnan Pillaipakkamnatt, Vijay Raghavan, and Dawn Wilkins. How many queries are needed to learn? Journal of the ACM, 43(5):840-862, September 1996.

[2] Dana Angluin, Negative Results for Equivalence Queries, Machine Learning, v.5 n.2, p.121-150, Jun. 1990

[3] Dana Angluin, Queries and Concept Learning, Machine Learning, v.2 n.4, p.319-342, April 1988

[4] Tibor Hegedüs, Generalized teaching dimensions and the query complexity of learning, Proceedings of the eighth annual conference on Computational learning theory, p.108-117, July 05-08, 1995, Santa Cruz, California, United States

[5] M. Kearns and U. Vazirani. An Introduction to Computational Learning Theory.

[6] S. Goldman and M. Kearns. On the Complexity of Teaching. Proceedings of the Fourth Annual Workshop on Computational Learning Theory, pages 303–314, 1991.

[7] Jos L. Balczar, Jorge Castro, David Guijarro: A New Abstract Combinatorial Dimension for Exact Learning via Queries. J. Comput. Syst. Sci. 64(1): 2-21 (2002)

[8] Jos L. Balczar , Jorge Castro , David Guijarro , Hans-Ulrich Simon, The consistency dimension and distribution-dependent learning from queries, Theoretical Computer Science, v.288 n.2, p.197-215, 16 October 2002