A new abstract combinatorial dimension for exact learning via queries[*]

José Luis Balcázar, Jorge Castro and David Guijarro

Abstract

We introduce an abstract model of exact learning via queries that can be instantiated to all the query learning models currently in use, while being closer to them than previous unificatory attempts. We present a characterization of those Boolean function classes learnable in this abstract model, in terms of a new combinatorial notion that we introduce, the abstract identification dimension. Then we prove that the particularization of our notion to specific known protocols such as equivalence, membership, and membership and equivalence queries results in exactly the same combinatorial notions currently known to characterize learning in these models, such as strong consistency dimension, extended teaching dimension, and certificate size. Our theory thus fully unifies all these characterizations. For models enjoying a specific property that we identify, the notion can be simplified while keeping the same characterizations. From our results we can derive combinatorial characterizations of all those other models for query learning proposed in the literature. We can also obtain the first polynomial-query learning algorithms for specific interesting problems such as learning DNF with proper subset and superset queries.

[*] Work supported in part by the EC through the Esprit Program EU BRA program under project 20244 (ALCOM-IT), the EC Working Group EP27150 (NeuroColt II) and the spanish PB98-0937-C04-04. Part of this work was done while the first author visited CRM.

1 Introduction

The main models of exact learning via queries were introduced by Angluin [1]. In these models, the learning algorithm obtains information about the target concept asking queries to a teacher or expert. The algorithm has to output an exact representation of the target concept in polynomial time.

A main issue in exact learning is to decide whether a class is learnable with a polynomial number of queries regardless of the computation time needed between one query and the next. If this is not the case, then we do not need to dedicate any extra effort to obtain a polynomial time algorithm. There have been various ways of addressing this problem for different types of queries [14, 15, 2, 7, 8, 11, 13, 3, 4, 10, 5]. However, none of them obtained a uniform combinatorial characterization, applicable to all query learning protocols, of the number of queries needed to learn, in a similar way to the Vapnik-Chervonenkis dimension in the PAC learning model. This paper presents a dimension that can be seen as the VCdim brother for the exact learning setting.

We now explain the chain of results that led to the present paper. A combinatorial notion, called approximate fingerprints, turned out to characterize precisely those concept classes that can be learned from polynomially many equivalence queries of polynomial size [2, 7]. The essential intuition behind that fact is that the existence of queries that shrink the number of possibilities for the target concept by an inverse polynomial factor is not only clearly sufficient, but also necessary to learn: if no such queries are available then adversaries can be designed that force any learner to spend too many queries in order to identify the target. This intuition can be fully formalized along the lines of the cited works; the formalization can be found in [10].

Hellerstein et al. gave a beautiful characterization of the learnability of a representation class from membership and equivalence queries [13]. They introduced the notion of polynomial certificates for a representation class $\mathcal{R}$ and proved that $\mathcal{R}$ is polynomially learnable from equivalence and membership queries iff it has polynomial size certificates. They also prove that, for projection-closed classes, the teaching dimension introduced previously by Goldman and Kearns [8] characterizes learnability from membership queries. By broadening the notion into the extended teaching dimension, sort of a maximum between teaching dimension and certificate size, Hegedüs [11] characterizes learnability from membership queries without the projection-closed condition.

In [5], a quantitative analysis of certificates is presented, yielding the consistency dimension (or certificate size), and obtaining a precise characterization in such terms of the number of queries needed to learn. A related notion, the strong consistency dimension, is introduced and proved to characterize learning from just equivalence queries, in a manner quite different (and also simpler to handle) than the approximate fingerprints.

Here we move into a somewhat more abstract framework, and prove that all three concepts, strong consistency dimension from [5], certificates from [13], and extended teaching dimension from [11], are just three incarnations of the same abstract phenomenon. Indeed, we characterize rather tightly in our abstract framework the number of queries needed to learn by means of our new combinatorial concept of *abstract identification dimension* (AIdim), and prove that its instantiation to each of the three models mentioned coincides with the known combinatorial dimension for the corresponding model; but, likewise, it yields combinatorial characterizations of learning from, e.g., subset queries, or each of the models proposed in [1], or projective equivalence queries from [12]. We also study some cases in which a natural but nontrivial property of the learning protocol allows us to simplify the characterization.

As a bonus, the understanding of how a learning algorithm may work for these protocols yields the first algorithms for learning DNF from proper subset and superset queries, or from proper projective equivalence queries, that we describe in Section 5.

2 Notation and the abstract setting for exact learning

We assume familiarity with the exact learning model via queries. We focus on exact learning of Boolean functions, as an extremely basic form of knowledge. We fix all along the paper n as the number of variables. A Boolean function of arity n is a function from $\{0,1\}^n \rightarrow \{0,1\}$. The set of all Boolean functions is denoted by B_n.

An element x of $\{0,1\}^n$ is called an *assignment*. A pair (x,b), where $b \in \{0,1\}$ is a binary label, is called *example for function* $f \in B_n$ if $f(x) = b$. A *sample*, also called a *partial function* or *partially defined concept*, is a collection of examples for some function $f \in B_n$, and can be seen equivalently as a function from $\{0,1\}^n$ to $\{0,1,\star\}$, where "$\star$" stands for "undefined". The set of all samples on n variables is denoted by $Sample_n$. Note that $B_n \subseteq Sample_n$. A sample a is said to be *consistent with sample* b, denoted $a \sqsubseteq b$, if $a(x) = b(x)$ whenever $a(x) \neq \star$. This notation is extended to $S \sqsubseteq H$, for sets of samples $S, H \subseteq Sample_n$, if $(\forall a \in S)(\exists b \in H)(a \sqsubseteq b)$. Observe that for $a \in Sample_n$ and $F \subseteq B_n$, $a \not\sqsubseteq F$ means that no function from F is consistent with a. For a sample $a \in \mathit{Sample}_n$, a^+ denotes the set $\{(x,1) \mid x \in \{0,1\}^n, a(x)=1\}$ and a^- is the set $\{(x,0) \mid x \in \{0,1\}^n, a(x)=0\}$. We denote by $\|X\|$ the cardinality of set X and by $A \oplus B$ the join (marked union) between sets A and B.

2.1 An abstract setting for queries and answers

In our abstract setting, queries are atomic objects. Answers provide some partial knowledge of the target. Since our target concepts are always Boolean functions, we assume that such partial knowledge is always modeled as the values of the target function on a subdomain; thus, each answer is just a partial Boolean function (or: a sample) that is a subfunction of the target (or: that is consistent with it). The queries that give this kind of answers are sometimes called example-based queries (see [8], for example).

Thus, starring any abstract learning protocol we have three participants: the set Q of queries, the set of all Boolean functions B_n of some arity n, and the set of all possible answers, namely all the *partial* Boolean functions of the same fixed arity n. Since the set of all Boolean functions and the arity n will be constants in our discourse, and the set of answers will be specifically defined by each learning protocol, we only write explicitly the dependence of the protocol in Q. A protocol $\mathit{Protocol}(Q)$ is a subset of

$$\{\langle q, f, a\rangle \mid q \in Q, f \in B_n, a \sqsubseteq f\}\,.$$

For instance, if we want to talk about learning with the usual equivalence queries with hypothesis coming from a subset $H \subseteq B_n$, we define $\mathit{Protocol}_{\equiv}(H)$ as the set

$$\{\langle h, f, h\rangle \mid h \in H, f \in B_n, h \equiv f\} \\ \cup \{\langle h, f, a\rangle \mid h \in H, f \in B_n, a \in (f^- - h^-) \cup (f^+ - h^+)\}$$

where the first set corresponds to YES answers and the second to counterexamples. In a similar way we can define the protocol for some of the other queries defined in [1]:

- For membership queries on $M \subseteq \{0,1\}^n$ the set $\mathit{Protocol}_{\in}(M)$ is $\{\langle x, f, (x, f(x))\rangle \mid x \in M,\ f \in B_n\}$.
- For membership queries on a set $M \subseteq \{0,1\}^n$ and equivalence queries on a set $H \subseteq B_n$ the set $\mathit{Protocol}_{\in,\equiv}(M \oplus H)$ is $\mathit{Protocol}_{\equiv}(H) \oplus \mathit{Protocol}_{\in}(M)$.
- For subset queries on a class $H \subseteq B_n$ the set $\mathit{Protocol}_{\subseteq}(H)$ is

$$\{\langle h, f, h^+\rangle \mid h \in H, f \in B_n, h^+ \subseteq f^+\} \\ \cup \{\langle h, f, a\rangle \mid h \in H, f \in B_n, a \in f^- - h^-\}\,.$$

- For superset queries on a class $H \subseteq B_n$ the set $\mathit{Protocol}_{\supseteq}(H)$ is

$$\{\langle h, f, h^- \rangle \mid h \in H, f \in B_n, h^- \subseteq f^-\} \\ \cup \{\langle h, f, a \rangle \mid h \in H, f \in B_n, a \in f^+ - h^+\}\,.$$

- For both subset on $A \subseteq B_n$ and superset queries on $B \subseteq B_n$, the set $\mathit{Protocol}_{\subseteq,\supseteq}(A \oplus B)$ is $\mathit{Protocol}_{\subseteq}(A) \oplus \mathit{Protocol}_{\supseteq}(B)$.

We need to impose some conditions on the protocol to capture the notion of exact learning. First, we will force that the protocol has legitimate answers for every allowed query under every Boolean function. Second, we will include a "fair play" condition, namely, answers give no extra information beyond what we intend to give with them.

Thus, an *abstract learning protocol* $P = \mathit{Protocol}(Q)$, from now on a *protocol*, must fulfill the following conditions:

1. **Completeness** For each $q \in Q$ and $f \in B_n$, there is at least one $a \sqsubseteq f$ such that $\langle q, f, a \rangle \in P$. In words, all queries must have at least one answer.
2. **Fair-play** If $\langle q, f, a \rangle \in P$ and $a \sqsubseteq h$ for some other $h \in B_n$, then $\langle q, h, a \rangle \in P$.

The fair play condition will be central to all of our work. We will find the proofs repeatedly resorting to that condition. Observe that if it does not hold for some $\langle q, f, a \rangle$ and h, then the answer a to query q would provide side information, allowing the learner to discard a target h even though it is consistent with the answer a received.

In some definitions we will be locally interested in considering *answering schemes*. We say that $T \subseteq P$ is an answering scheme for a protocol P when T fulfills the completeness condition. Note that the protocol P is also an answering scheme. For an answering scheme T, we denote by $T^f(q) = \{a \mid \langle q, f, a \rangle \in T\}$, the set of potential answers to query q under function f, and by $T^f = \{a \mid \exists q \in Q\, \langle q, f, a \rangle \in T\}$, the set of all potential answers under function f, which coincides with $\bigcup_{q \in Q} T^f(q)$.

2.2 Exact learning

We use a generalization of the exact learning model via queries of Angluin [1]. A teacher answers with respect to $f \in B_n$ and using $P = \mathit{Protocol}(Q)$ if for each query $q \in Q$, it outputs some $a \in P^f(q)$. A function class $C \subseteq B_n$ is learnable with d queries under $P = \mathit{Protocol}(Q)$ if there exists an algorithm A such that for any $f \in C$ and for any teacher B that answers with respect to f using P, the only remaining function in C that is consistent with the answers received after at most d interactions is f. For a class $C \subseteq B_n$ and a protocol $P = \mathit{Protocol}(Q)$ we define the *learning complexity*, $\mathrm{LC}(C, P)$, as the smallest d such that C is learnable with d queries under P.

We define the notion of a version space that will be useful for the learning algorithms that we use in all the paper. At any intermediate stage of a query-learning process, the learner knows (from the teacher's answers received so far) a set of samples S for the target concept. Let C be the target class. The *version space* $\mathcal{V}$ is the set of all concepts from C which are consistent with all samples in S. These are all concepts being still conceivable as target concepts.

A fully general, rather simple way of extracting a combinatorial parameter from an abstract learning protocol is to use a chain of alternating quantifiers of queries and answers. We describe it here, as a way of introducing the idea, and also for the sake of comparison with the much nicer "flat" version we will describe in the next section; it will be also useful for technical purposes in a later proof.

Given a class $C \subseteq B_n$ and a protocol $P = \mathit{Protocol}(Q)$, the *ugly dimension*, $\mathrm{Udim}(C, P)$, is the minimum integer d such that for any $f \in B_n$ (not just in C!)

$$(\exists q_1 \in Q)(\forall a_1 \in P^f(q_1)) \dots (\exists q_d \in Q)(\forall a_d \in P^f(q_d))$$
$$(\|\{c \in C \mid \{a_1, \dots, a_d\} \sqsubseteq c\}\| \leq 1)$$

if no such d exists then $\mathrm{Udim}(C, P) = \infty$.

Now, using fully standard techniques, we can easily prove the following theorem.

Theorem 1. *For any class $C \subseteq B_n$ and any protocol $P = \mathit{Protocol}(Q)$*

$$\mathit{Udim}(C, P) \leq \mathit{LC}(C, P) \leq \mathit{Udim}(C, P) \lceil \log \|C\| \rceil .$$

Proof. If $\mathrm{Udim}(C, P) > k$ then there exists $f \in B_n$ such that

$$(\forall q_1 \in Q)(\exists a_1 \in P^f(q_1)) \dots (\forall q_k \in Q)(\exists a_k \in P^f(q_k))$$
$$(\|\{c \in C \mid (\{a_1, \dots, a_k\} \sqsubseteq c)\}\| > 1)$$

which describes an adversary that can force any learner to make more than k queries.

On the other side, assume $\mathrm{Udim}(C, P) \leq k$ and let $\mathcal{V}$ be the version space in an intermediate step of the learning algorithm that we are now describing (initially $\mathcal{V} = C$). Let $f_{\mathcal{V}}$ be the majority function on $\mathcal{V}$, i.e.

$f_{\mathcal{V}}(x) = 1$ if more than $\frac{1}{2}$ of the functions in $\mathcal{V}$ classify x as 1, or $f_{\mathcal{V}}(x) = 0$ otherwise. The bound on $\mathrm{Udim}(C, P)$ promises that there exists a query q_1 such that for all answers a_1 labelled according to $f_{\mathcal{V}}$, and so on and so forth, there is at most one function in C that is consistent with all those answers. Therefore we run the process of asking $q_1 \dots q_k$ (q_{i+1} depends on the previous answers). If all answers are consistent with $f_{\mathcal{V}}$ then, by the fair play property, they all belong to $P^{f_{\mathcal{V}}}$ and there is only one function in C consistent with them, the target. Otherwise, at least $\frac{1}{2}$ of the functions in $\mathcal{V}$ are discarded and we start again with $\mathcal{V}$ half the size as before. This process is repeated at most $\lceil \log \|C\| \rceil$ times. □

In the next section we present a nicer dimension that does not need alternating quantifiers and also gives an approximation (in the same sense as Theorem 1) to the number of queries needed to learn.

3 The abstract identification dimension

Given a target class $C \subseteq B_n$ and a protocol $P = \mathit{Protocol}(Q)$, we define the *abstract identification dimension*, $\mathrm{AIdim}(C, P)$, as the minimum integer d such that

$$(\forall f \in B_n)\,(\forall T \in \mathcal{T}(P))\,(\exists S \subseteq T^f)\,(\|S\| \leq d \wedge \|\{h \in C \mid S \sqsubseteq h\}\| \leq 1)\,.$$

If no such integer exists then $\mathrm{AIdim}(C, P) = \infty$.

That is, no matter what Boolean function and answering scheme are chosen there exists some set of at most d answers such that at most one function in the target class is consistent with those answers.

The following lemma will be central in the proof of our main result in this section and is interesting in its own right.

Lemma 2. *Let $C \subseteq B_n$, $D \subseteq C$ such that $\|D\| > 1$, $P = \mathit{Protocol}(Q)$, $\mathit{AIdim}(C, P) = d$ and f be any function in B_n. There exists $q \in Q$ such that for any $a \in P^f(q)$, at least $\frac{\|D\|}{2d}$ functions from D are inconsistent with some assignment in a.*

Proof. For the sake of contradiction suppose that for each $q \in Q$ there exists some $a_q \in P^f(q)$ such that less than $\frac{\|D\|}{2d}$ functions are inconsistent with some assignment in a_q. Then we define an answering scheme T such that $T^f(q) = \{a_q\}$. Now for any $S \subseteq T^f$ such that $\|S\| \leq d$ there are less than $\frac{d\|D\|}{2d}$ functions inconsistent with some assignment in S which implies that there must be at least two functions in D that are consistent with S. This contradicts $\mathrm{AIdim}(C, P) = d$. □

Our main contribution of this section is the following characterization:

Theorem 3. *For any concept class $C \subseteq B_n$ and any protocol $P = Protocol(Q)$,*

$$AIdim(C,P) \leq LC(C,P) \leq 2AIdim(C,P)\lceil \ln \|C\| \rceil \, .$$

Proof. We will start showing that if $\text{AIdim}(C,P) > k$ then any learning algorithm must ask more than k queries. For the sake of contradiction suppose that there is an algorithm A that learns C asking at most k queries. Let f and T be the Boolean function and the answering scheme such that

$$(\forall S \subseteq T^f)(\|S\| \leq k \Rightarrow \|\{h \in C \mid S \sqsubseteq h\}\| > 1)$$

obtained by negation of the definition of AIdim.

Now we answer all queries from A using T. After k interactions, A knows a set of given answers $S_A \subseteq T^f$, and by the choice of T and f, there exist two different functions in C that are consistent with all assignments in S_A. This contradicts the assumption on A. Observe that even though f is not necessarily in C it can be claimed that the answers were given according to one of the two surviving functions from C because of the fair play property.

Now we show the upper bound. Assume $\text{AIdim}(C,P) \leq k$. Let $\mathcal{V}$ be the version space consisting of functions in C that are consistent with the answers received so far (initially $\mathcal{V} = C$). Let $f_{\mathcal{V}}$ be the majority function on $\mathcal{V}$. Now we make the query whose existence is guaranteed by Lemma 2. If the answer is inconsistent with $f_{\mathcal{V}}$ then at least $\frac{1}{2}$ of the functions in $\mathcal{V}$ are removed, otherwise the answer is in $P^{f_{\mathcal{V}}}$ (because of the fair play property) and therefore Lemma 2 ensures that $\frac{\|\mathcal{V}\|}{2k}$ functions from $\mathcal{V}$ are inconsistent with some assignment in the answer received. An easy counting argument finishes the proof of the theorem. □

Next we show a necessary and sufficient condition for $\text{AIdim}(C,P)$ being ∞.

Theorem 4. *For any $C \subseteq B_n$ and for any $P = Protocol(Q)$, $AIdim(C,P) \neq \infty$ if and only if for all $f, g \in C$, such that $f \neq g$ there exists $q \in Q$, $P^f(q) \cap P^g(q) = \emptyset$. Furthermore, if $AIdim(C,P) \neq \infty$ then $AIdim(C,P) \leq \|C\| - 1$.*

Proof. Suppose that for all $f, g \in C$, $f \neq g$, there exists some $q \in Q$ such that $P^f(q) \cap P^g(q) = \emptyset$. Then it is easy to design an algorithm that makes at most $\|C\| - 1$ queries: it takes a pair of functions from C, asks the separating query and for any answer of the teacher at least one of the two functions is discarded (again by the fair play). This implies that $\text{AIdim}(C,P) \leq \|C\| - 1$ because of Theorem 3.

Conversely, assume that there exist $f, g \in C$, $f \neq g$ such that for all $q \in Q$, $P^f(q) \cap P^g(q) \neq \emptyset$ and call those witnesses of the nonempty intersection $a_{f,g,q}$. Let T be an answering scheme such that $T^f(q) = \{a_{f,g,q}\}$. Observe that for all $S \subseteq T^f$ both f and g are consistent with S which implies that $\mathrm{AIdim}(C, P) = \infty$. □

We prove now that $\mathrm{AIdim}(C, P)$ corresponds with the dimension introduced in [5] for the case of equivalence queries: the strong consistency dimension. For a target class $C \subseteq B_n$ and a class Q ($C \subseteq Q \subseteq B_n$) of hypothesis for the equivalence queries, the *strong consistency dimension*, $\mathrm{scdim}(C, Q)$, can be written as the minimum integer d such that

$$(\forall g \in \mathit{Sample}_n)(g \not\sqsubseteq Q \Rightarrow (\exists S \sqsubseteq g)(\|S\| \leq d \wedge S \not\sqsubseteq C))\,.$$

The following result relates, rather tightly, both dimensions.

Theorem 5. *For any $C \subseteq B_n$ and protocol $P = \mathit{Protocol}_{\equiv}(Q)$ such $C \subseteq Q$,*

$$\mathit{AIdim}(C, P) \leq \mathit{scdim}(C, Q) \leq \mathit{AIdim}(C, P) + 1\,.$$

Proof. Let $d_s = \mathrm{scdim}(C, Q)$ and $\mathrm{AIdim}(C, P) = d_a$. Observe that any sample $g \not\sqsubseteq Q$ provides with all the information needed to build an answering scheme in the case of equivalence queries.

For the first inequality, let f be any function from B_n. There are two cases: (a) $f \in Q$ and (b) $f \in B_n - Q$. In case (a) one single answer suffices to rule out all but one functions in C, namely the unique answer to f itself provides all f as answer and only one function can be consistent with that. For case (b) consider any answering scheme for f, T. Observe that T^f can be seen as a sample $g \sqsubseteq f$ such that $g \not\sqsubseteq Q$. We use the scdim(,) machinery: $(\exists S \sqsubseteq g)(\|S\| \leq d_s \wedge S \not\sqsubseteq C)$ which implies that $d_s \geq d_a$.

For the second inequality let $g \in \mathit{Sample}_n$ be such that $g \not\sqsubseteq Q$ (and therefore $g \not\sqsubseteq C$). Now consider any total function f in B_n such that $g \sqsubseteq f$ and an answering scheme T such that $T^f \sqsubseteq g$. Now we know that there exists some $S \subseteq g$, of size at most d_a such that at most one function in C is consistent with it. If there is one such $c \in C$ we add one more example from $(g^+ - c^+) \cup (g^- - c^-)$ to S and then rule out all possible functions from C. □

The next section will prove that, under an additional condition on P, the definition of $\mathrm{AIdim}(C, P)$ can be simplified, and will show how it corresponds to known characterizations of other learning protocols.

4 Enforcing answers

Many learning protocols (but not all, the most notable exception being equivalence queries) have the following property: for each potential answer (in our abstract sense), there is some query that enforces exactly that answer. A simple example is related to membership queries: an answer consisting of a labeled example (or: sample of size 1) can be taken as counterexample as one among many answers to an equivalence query, but is the only possible answer to a membership query. The purpose of this section is to show that it is exactly this property the key to the differences between known characterizations of query learning protocols.

We say that the abstract learning protocol P has the *enforcing answers property* if, for each $\langle q, f, a\rangle \in P$, there is a query q' such that $P^f(q') = \{a\}$. That is, for each potential answer, some possibly different query forces it as the only authorized answer.

Our main result of this section says that, under this extra condition, one can dispose of considering all answering schemes in the definition of abstract identification dimension. We define the *enforcing abstract identification dimension*, $\mathrm{EAIdim}(C, P)$, as the smallest integer d such that

$$(\forall f \in B_n)(\exists S \subseteq P^f)(\|S\| \leq d \wedge \|\{h \in C \mid S \sqsubseteq h\}\| \leq 1)\,.$$

If there is no such d then $\mathrm{EAIdim}(C, P) = \infty$.

Theorem 6. *Let $C \subseteq B_n$ and $P = Protocol(Q)$. If P has the enforcing answer property then,*

$$EAIdim(C, P) = AIdim(C, P) = Udim(C, P)\,.$$

Proof. Clearly $\mathrm{EAIdim}(C, P) \leq \mathrm{AIdim}(C, P)$ because P is itself an answering scheme. It is also easy to see that $\mathrm{AIdim}(C, P) \leq \mathrm{Udim}(C, P)$. Observe that the two previous facts are independent of the enforcing answers property.

To prove $\mathrm{Udim}(C, P) \leq \mathrm{EAIdim}(C, P)$ we need the enforcing answers property. Note that $\mathrm{EAIdim}(C, P) = d$ can be interpreted as follows: for any $f \in B_n$, d answers, $\{a_1, \ldots, a_d\} \subseteq P^f$, suffice to eliminate all but one functions from C. Since any answer a_i has a query q_i such that $P^f(q_i) = \{a_i\}$, then for any $f \in B_n$

$$\begin{aligned}(\exists q_1 \in Q) \ldots (\exists q_d \in Q)(\forall a_1 \in P^f(q_1)) \ldots (\forall a_d \in P^f(q_d))&\\ (\|\{c \in C \mid \{a_1, \ldots, a_d\} \sqsubseteq c\}\| \leq 1)&\end{aligned}$$

and therefore, $\mathrm{Udim}(C, P) \leq d$. □

After Theorems 1 and 6, for each $C \subseteq B_n$, $P = \mathit{Protocol}(Q)$ such that P has the enforcing answers property, one has $\mathrm{AIdim}(C,P) \leq \mathrm{LC}(C,P) \leq \mathrm{AIdim}(C,P)\lceil \log \|C\| \rceil$. One may wonder whether this gap of $\lceil \log \|C\| \rceil$ could be improved. The next easy to prove theorem shows that this is not so easy for general classes and protocols, since there are examples of having the equality on both ends of the gap. Let SING_n be the class of singleton functions on n variables.

Theorem 7. *For $P = \mathit{Protocol}_{\in}(\{0,1\}^n)$, $\mathit{AIdim}(\mathit{SING}_n, P) = \mathit{LC}(\mathit{SING}_n, P)$ and for $P = \mathit{Protocol}_{\equiv}(B_n)$, $\mathit{LC}(B_n, P) = \mathit{AIdim}(B_n, P) \log \|B_n\|$.*

The simplification introduced by Theorem 6 allows us to prove that the abstract identification dimension generalizes two more characterizations of learning protocols: the certificate size for membership and equivalence queries, and the extended teaching dimension for just membership queries, in the same way as we proved in the previous section that it generalizes the strong consistency dimension for equivalence queries.

The certificate size in [13] (or *consistency dimension* in [5]) of a target class $C \subseteq B_n$ and a hypothesis class $H \subseteq B_n$, $\mathrm{cdim}(C,H)$, is the smallest integer d such that

$$(\forall f \in B_n)(f \not\sqsubseteq H \Rightarrow (\exists s \sqsubseteq f)(\|s\| \leq d \wedge s \not\sqsubseteq C))$$

or ∞ if no such d exists.

Theorem 8. *For any $C, H \subseteq B_n$, $C \subseteq H$, and $P = \mathit{Protocol}_{\in,\equiv}(\{0,1\}^n \oplus H)$*

$$\mathit{AIdim}(C,P) \leq \mathit{cdim}(C,H) \leq \mathit{AIdim}(C,P) + 1$$

Proof. The proof follows similar steps to the proof of Theorem 5. □

The *extended teaching dimension* [11] (see also [13]) of some class $C \subseteq B_n$, $\mathrm{etdim}(C)$, is the smallest integer d such that

$$(\forall f \in B_n)(\exists s \sqsubseteq f)(\|s\| \leq d \wedge \|\{c \in C \mid s \sqsubseteq c\}\| \leq 1)$$

or ∞ if no such d exists.

The following theorem is immediate.

Theorem 9. *For any $C \subseteq B_n$ and $P = \mathit{Protocol}_{\in}(\{0,1\}^n)$, $\mathit{AIdim}(C,P) = \mathit{etdim}(C)$*

We end this section with an example showing that $\mathrm{EAIdim}(C,P)$ is not valid in general as an approximation of the number of queries needed for exact learning when the enforcing answers property does not hold. Let C be the class of k-term monotone DNF, for some constant $k > 1$, and P be $\mathit{Protocol}_{\equiv}(C)$. It is easy to see that $\mathrm{EAIdim}(C,P) \leq (k+1)(n+1)$ but $\mathrm{AIdim}(C,P)$ is not bounded by any polynomial in n (see [6], for example).

5 Applications

Our setting immediately provides with new combinatorial characterizations of all other popular learning protocols and with the first exact learning algorithm for DNFs that uses polynomially many queries that are DNFs of polynomial size. We start, as an example, with subset queries and then move to the algorithms for learning DNF formulas.

5.1 Subset queries

We need some definitions specific for subset queries. For a sample $g \in \mathit{Sample}_n$, we say that g is *valid* for $H \subseteq B_n$ if and only if $\forall h \in H$ either $h^+ \sqsubseteq g$ or $h^+ - g^+ \neq \emptyset$. The *covering cost* of a sample g with H is $\mathrm{covcost}_H(g) = \|g^-\| + \min\{j \mid (\exists h_1 \ldots h_j)(\bigcup h_i^+ = g^+)\}$. If such j does not exist then $\mathrm{covcost}_H(g) = \infty$. We denote by $\mathrm{covdim}(C, H)$ the smallest integer d such that

$$(\forall g \in \mathit{Sample}_n)(g \textit{ is valid for } H \Rightarrow (\exists s \sqsubseteq g)(\mathrm{covcost}_H(s) \\ \leq d \wedge \|\{c \in C \mid s \sqsubseteq c\}\| \leq 1))$$

or ∞ if no such d exists.

Theorem 10. *For any pair of classes $C, H \subseteq B_n$ and $P = \mathit{Protocol}_{\subseteq}(H)$, $\mathit{AIdim}(C, P) = \mathit{covdim}(C, H)$.*

Proof. It is enough to observe that a sample being valid for H corresponds to the notion of answering scheme and that the covcost() function measures the minimum number of answers to subset queries contained in a sample. □

5.2 Learnability of DNF formulas and related classes

All the intuitions gleaned through this work have more specific applications, in particular by illuminating how query learning algorithms might proceed using the powerful subset and superset queries, or the less known projective equivalence queries of [12]. Next subsection deals with that.

We need some more definitions. A partial assignment α is a word from $\{0, 1, \star\}^n$. A complete assignment $x \in \{0, 1\}^n$ satisfies a partial assignment α if they coincide in the positions where α is not $\star$. The hypercube of a partial assignment α is the set of all complete assignments that satisfy α. We denote by $t(\alpha)$ the term that, when applied to a complete assignment x, evaluates to 1 if x satisfies α and to 0 otherwise and by $c(\alpha)$ the clause such that $c(\alpha) = \bar{t}(\alpha)$. A function $f \in B_n$ projected with respect to α is denoted by f_α. The function f_α is equal to $t(\alpha) \wedge f$. Observe that our definition is not the projection of [13] because the number of variables is not reduced.

The following theorem states the first known exact learning result for DNF formulas that uses a polynomial number of queries of polynomial size that are also DNF formulas.

Theorem 11. *The class of DNF formulas with at most m terms and over n variables is learnable with $2nm\lceil\log 3\rceil$ subset and superset queries that are DNF formulas with at most $2m+n$ terms.*

Proof. Assume, w.l.o.g, that $m \geq 1$. Let G be the class of DNF formulas with at most $2m$ terms, H be the class of DNF formulas with at most $2m+n$ terms, C be the class of DNF formulas with at most m terms and $P = \mathit{Protocol}_{\subseteq,\supseteq}(H \oplus H)$. Observe that $C \subseteq G \subseteq H$. Since the enforcing answers property applies, it is enough to show that $\mathrm{AIdim}(C, P) = 2$ (which coincides with $\mathrm{EAIdim}(C, P)$ because of Theorem 6) and the theorem follows because of Theorem 1 and the fact that $\log \|C\| \leq nm\lceil\log 3\rceil$.

Consider any function $f \in B_n$. There are two cases: (a) $f \in G$ and (b) $f \notin G$. In case (a) the answers in P^f to two queries suffice to discard all but one functions in B_n, namely the answers to subset on f and superset on f. In case (b) we use a projection trick: we project f according to some partial assignment α (initially $\alpha = \star^n$) while for any variable v not yet projected there exists a Boolean value b such that $f_{\alpha\cup v\leftarrow b} \notin G$, we choose any such variable and value and continue projecting. Since both SING_n and the constant 0 are in G we have to reach some point where we have projected according to some partial assignment α such that $f_\alpha \notin G$ and there exists some variable v such that both $f_{\alpha\cup v\leftarrow 0}$ and $f_{\alpha\cup v\leftarrow 1}$ are in G. Now, because $f_\alpha = f_{\alpha\cup v\leftarrow 0} \vee f_{\alpha\cup v\leftarrow 1}$, at least one of the two projections must be outside C, otherwise f_α would be in G. Therefore there exists $b \in \{0, 1\}$ such that $f_{\alpha\cup v\leftarrow b} \notin C$. Let β be $\alpha \cup v \leftarrow b$. Now, the unique answers according to P^f, on subset of f_β and superset of $f_\beta \vee c(\beta)$ (that both belong to H) give all the hypercube that satisfies β labelled according to f. Since $f_\beta \notin C$ those examples discard all functions from C because C is projection closed. □

Now we prove a similar result using the less known projective equivalence queries from [12]. A projective equivalence query receives as input a partial assignment α and a hypothesis $h \in B_n$ and the answer is the hypercube that satisfies α if h and the target are consistent there or some example in that hypercube witnessing the fact that they do not coincide.

Using similar arguments to the proof of Theorem 11 we can prove the following result.

Theorem 12. *The class of DNF formulas with at most m terms and over n variables is learnable with $nm\lceil\log 3\rceil$ projective equivalence queries that are DNF formulas with at most $2m$ terms.*

Proof. In this case it can be shown that $\mathrm{AIdim}(C, P) = 1$. □

In fact, the only properties of DNFs employed in the previous results are:

1. If the number of terms needed to represent $f \in B_n$ in DNF form is more than $2m$ then for any variable v there exists a Boolean value b, such that $f_{v \leftarrow b}$ needs more than m terms.
2. Given a Boolean function f representable as a DNF with at most m terms and a clause c on n variables, the function $f \vee c$ can be represented with at most $m + n$ terms.
3. The class of SING_n is representable as DNF with at most 1 term and DNF formulas with at most $m \geq 1$ terms are projection closed.

Since those properties are also satisfied by Decision Trees, Branching Programs, Decision Lists and Boolean Formulas (with some minor variations in the numbers that are still within a polynomial), the same result holds for them.

Corollary 13. *Decision Trees, Decision Lists, Branching Programs and Boolean Formulas are learnable with a polynomial number of queries of polynomial size both with proper subset and superset queries or with proper projective equivalence queries.*

Observe that subset and superset queries together and also projective equivalence queries can simulate the membership and equivalence queries protocol for the classes considered above. Since for DNF formulas and Decision Trees it is known that membership queries or proper equivalence queries do not suffice (see [2, 9]), the case of using both membership queries and proper equivalence queries (an important open problem) fall now between by the positive results in this paper with more powerful queries and the negative results for weaker protocols.

References

[1] D. Angluin. Queries and concept learning. *Machine Learning*, 2:319–342, 1988.

[2] D. Angluin. Negative results for equivalence queries. *Machine Learning*, 5:121–150, 1990.

[3] V. Arvind and N. V. Vinodchandran. The complexity of exactly learning algebraic concepts. In *Algorithmic Learning Theory: ALT '96*, pages 100–112. Springer-Verlag, 1996.

[4] V. Arvind and N. V. Vinodchandran. Exact learning via teaching assistants. In *Algorithmic Learning Theory: ALT '97*, pages 291–306. Springer-Verlag, 1997.

[5] J. L. Balcázar, J. Castro, D. Guijarro, and H. U. Simon. The consistency dimension and distribution-dependent learning from queries. In *ALT'99*, volume 1720, pages 77–92. LNAI. Springer, 1999.

[6] J. Castro, D. Guijarro, and V. Lavín. Learning nearly monotone k-term dnf. *Information Processing Letters*, 67:75–79, 1998.

[7] R. Gavaldà. On the power of equivalence queries. In *Computational Learning Theory: Eurocolt '93*, pages 193–203. Oxford University Press, 1994.

[8] S. Goldman and M. Kearns. On the complexity of teaching. *Journal of Computer and System Sciences*, 50:20–31, 1995.

[9] D. Guijarro, V. Lavín, and V. Raghavan. Exact learning when irrelevant variables abound. *Information Processing Letters*, 70:233–240, 1999.

[10] Y. Hayashi, S. Matsumoto, A. Shinoara, and M. Takeda. Uniform characterization of polynomial-query learnabilities. In *First International Conference, DS'98*, volume 1532, pages 84–92, Fukuoka, Japan, 1998. LNAI. Springer.

[11] T. Hegedüs. Generalized teaching dimension and the query complexity of learning. In *8th Annu. Conf. on Comput. Learning Theory*, pages 108–117, New York, 1995. ACM Press.

[12] L. Hellerstein and M. Karpinsky. Learning read-once formulas using membership queries. In *2nd Annu. Conf. on Comput. Learning Theory*, pages 146–161, Palo Alto, 1989. Morgan Kaufmann.

[13] L. Hellerstein, K. Pillaipakkamnatt, V. Raghavan, and D. Wilkins. How many queries are needed to learn? *Journal of the ACM*, 43(5):840–862, 1996.

[14] W. Maass and G. Turán. On the complexity of learning from counterexamples. In *Proc. 30th Annu. IEEE Sympos. Found. Comput. Sci.*, pages 262–267. IEEE Computer Society Press, Los Alamitos, CA, 1989.

[15] W. Maass and G. Turán. On the complexity of learning from counterexamples and membership queries. In *Proc. of the 31st Symposium on the Foundations of Comp. Sci.*, pages 203–210. IEEE Computer Society Press, Los Alamitos, CA, 1990.

J. L. Balcázar, J. Castro and D. Guijarro
Jordi Girona Salgado, 1–3
Campus Nord,
LSI, UPC
08034 Barcelona, Spain
email:{balqui|castro|david}@lsi.upc.es