Query, PACS and simple-PAC Learning*

Jorge Castro and David Guijarro
Dept. Llenguatges i Sistemes Informàtics,
UPC, Campus Nord, Mòdul C5
Jordi Girona Salgado, 1-3, 08034 Barcelona, Spain
Email:{castro,david}@lsi.upc.es

January 30, 1998

Abstract

We study a distribution dependent form of PAC learning that uses probability distributions related to Kolmogorov complexity. We relate the PACS model, defined by Denis, D'Halluin and Gilleron in [3], with the standard simple-PAC model and give a general technique that subsumes the results in [3] and [6].

1 Introduction

One of the most relevant models of learning is "PAC learning", introduced by Valiant [7], which has been widely used to investigate the phenomenon of learning from examples. Informally, in this model one has to learn a target concept with high probability in polynomial time (and, a fortiori, from a polynomial number of examples), within a certain error, under all probability distributions on the examples. Because of this last requirement, to learn under all distributions, PAC learning is also called distribution-free, or distribution-independent learning.

Distribution-independent learning is a strong requirement. Many concept classes are not known to be polynomially learnable, or known not to be polynomially learnable if RP $\neq$ NP, although some such concept classes are polynomially learnable under some fixed distributions. Li and Vitányi proposed in [4] the simple-PAC learning model that, roughly speaking, replaces the condition of learning under all distributions of Valiant's original model by the request of learning under all simple distributions, provided the sample is given according to the "universal" distribution.

*Supported by the Esprit Long Term Research Project ALCOM IT (nr. 20244), the Working Group 8556 (Neurocolt), and the Spanish DGICYT (project KOALA BP95-0797).

Simple distributions form a wide class that properly includes all enumerable ones. Specifically, they are those multiplicatively dominated by the universal enumerable distribution $\mathbf{m}$. This distribution assigns high probabilities to low Kolmogorov complexity examples. As we will see later, these properties of $\mathbf{m}$ have nice consequences with regard to PAC learning under simple distributions.

With the new model, Li and Vitányi [4] developed a theory of learning for simple concepts —concepts with low Kolmogorov complexity— that intuitively should be polynomially learnable. In fact, they showed several examples that strengthen this intuition.

Recently, Denis et al. in [3] have restricted this model to the case where a benign teacher might choose examples based on the knowledge of the target concept. In this framework, called PACS model, examples with short descriptions with respect to the target concept have high probability to be drawn. We show here that any query learnable class is also learnable under this extended framework. So, for example, learnability of DFA's in the PACS model (shown in [6]) is a consequence of its query learnability [1].

We also study the relationships between simple-PAC and PACS. We show that they are correlated if some conditions hold. These conditions are not unreasonable since they hold for all known algorithms. An interesting consequence of our results is that query learnability of any class implies simple-PAC learning of its simple concepts.

Section 2 is devoted to definitions and previous results, Section 3 presents the relation between query learning and PACS and Section 4 relates PACS and simple-PAC.

2 Preliminaires

For definitions on learning-from-examples, PAC/query learning and representation classes we refer the reader to [2, 7, 8]. We use Σ for the alphabet that codifies examples and $l(r)$ to denote, in a fixed representation class, the length of the shortest representation for a concept r in that class. The simmetric difference of sets A and B is denoted by $A \triangle B$.

A probability distribution P is called enumerable, if there exists a recursive function $g(x,k)$, nondecreasing in k, with $P(x) = \lim_{k\to\infty} g(x,k)$. It can be proved the existence of universal distributions in the class of enumerable distributions. That means that there are enumerable distributions that multiplicatively dominate each enumerable distribution. It can be shown that one of the universal enumerable distributions, denoted by $\mathbf{m}$, has the following properties

$$\sum_x \mathbf{m}(x) < 1,$$

and

$$\mathbf{m}(x) = 2^{-K(x)+O(1)},$$

where $K(x)$ denotes the Kolmogorov complexity and the equality is known as the Coding Theorem. The universal distribution has many important properties. Under $\mathbf{m}$, easily describable objects have high probability, and complex objects have low probability.

These results can be extended using the conditional Kolmogorov complexity $K(x|y)$ instead of $K(x)$ to define the universal conditional enumerable distribution $\mathbf{m}(x|y)$ that verifies

$$\forall y \in \Sigma^*, \sum_x \mathbf{m}(x|y) < 1,$$

and

$$\forall x, y \in \Sigma^*, \mathbf{m}(x|y) = 2^{-K(x|y)+O(1)},$$

where the equality is known as the Conditional Coding Theorem (see [5]).

Definition 2.1 *A distribution D is simple iff it is multiplicatively dominated by the universal distribution. That is, there exists a constant c, such that for all x,*

$$c\mathbf{m}(x) \geq D(x).$$

It can be shown that simple distributions properly include enumerable ones and that there is a distribution which is not simple. The following theorem relates learning under simple distributions with learning under the universal distribution $\mathbf{m}$.

Theorem 2.2 ([4]) *A concept class $\mathcal{C}$ is PAC learnable under the universal distribution $\mathbf{m}$, iff it is PAC learnable under all simple distributions, provided that in the learning phase the set of examples is drawn according to $\mathbf{m}$.*

In [4] it is shown how to exploit this completeness theorem to obtain new learning algorithms for DNF with simple terms and simple reversible languages.

Recently, Denis, D'Halluin and Gilleron [3] introduced the PACS model where the learning system might be aided by a benign teacher who knows the target concept and uses this knowledge in selecting the examples. Under this model, examples with low conditional Kolmogorov complexity have high probability. Formally, the probability of drawing an example x for a target concept with representation r is given as $\mathbf{m}_r(x) = \mu_r 2^{-K(x|r)}$ where μ_r satisfies $\mu_r \sum_x 2^{-K(x|r)} = 1$. By Kraft's inequality it holds that $\mu_r \geq 1$ (see [5]). Concept classes such as poly-term DNF and k-reversible DFA were shown to be learnable under the PACS model [3]. In a later work [6] the learnability of DFA's was also shown.

Note that $\mathbf{m}_r$ and $\mathbf{m}_\lambda$, where λ denotes the empty word, are similar to the normalized counterparts of the enumerable distributions $\mathbf{m}(\cdot|r)$ and $\mathbf{m}$. It is easy to see that all $\mathbf{m}_r$ are simple and universal for the class of simple distributions, but no one of distributios $\mathbf{m}_r$ can be enumerable (see Chapter 4 of [5]). We use in this paper as universal distributions $\mathbf{m}_r$ and $\mathbf{m}_\lambda$ instead of $\mathbf{m}(\cdot|r)$ and $\mathbf{m}$ (as Li and Vitányi proposed) in order to follow the choices in [3] and [6]. All the results below can be shown also if $\mathbf{m}$ and $\mathbf{m}(\cdot|r)$ are the choices considered and some of them can be obtained in an easier way. We note that Theorem 2.2 above remains true changing $\mathbf{m}$ by $\mathbf{m}_\lambda$. Now, we define simple-PAC

Definition 2.3 *A concept class $\mathcal{C}$ is simple-PAC learnable iff it is PAC learnable under $\mathbf{m}_\lambda$.*

3 From query learning to PACS learning

We show below that any class exactly learnable by an algorithm that uses queries is also learnable in the PACS model. We prove the result first for membership and equivalence queries and discuss after to what kind of queries our proof technique extends.

We restrict our attention only to query learning in the presence of bounded teachers. These teachers provide counterexamples up to a given length n or reply affirmatively if no counterexample is of length at most n, see [8] for a complete discussion. This is done for technical convenience only.

Let $\mathcal{R}$ be a representation class of a concept class $\mathcal{C}$, and let L be an exact learning algorithm for $\mathcal{R}$ that uses membership and equivalence queries. So, for any $r \in \mathcal{R}$, any positive integer n, and any bounded teacher T, algorithm L asks at most a polynomial number of queries $p(l(r), n)$ and outputs a representation $L^T(r) \in \mathcal{R}$ such that $L^T(r)$ is equivalent to r in $\Sigma^{\leq n}$. In the following we fix a teacher T_{min}; such teacher answers all the equivalence queries with the lexicographically smallest counterexample (if there exists any at all).

Lemma 3.1 *Fixed the target concept r and the teacher T_{min}, all membership queries and all counterexamples in the computation of L have Kolmogorov complexity $K(\cdot|r)$ bounded by $O(\log nl(r))$.*

Proof Let A_r be a fixed algorithm that given any hypothesis h, finds the lexicographically smallest example in $h \triangle r$ whenever $h \triangle r \neq \emptyset$. We enumerate all the membership queries and counterexamples in the computation of L with teacher T_{min}. By hypothesis, this enumeration ranges up to at most $p(l(r), n)$. So, any membership or counterexample word q can be described by means of its number in the list of queries, any representation of the target concept r, and algorithms L and A_r. Therefore, the Kolmogorov complexity of q conditional to r, $K(q|r)$, is bounded by $\log p(l(r), n) + K(r|r) + |L| + K(A_r|r) + O(1)$. Since $K(r|r)$, $|L|$, and $K(A_r|r)$ are constants, we obtain the bound of the lemma. $\Box$

Let $S_d^r(n)$ be the set $\{x \in \Sigma^{\leq n} : \mathbf{m}_r(x) \geq 1/n^d\}$. Note that $S_d^r(n)$ includes all the words $x \in \Sigma^{\leq n}$ such that $K(x|r) \leq d \log n$. The following standard lemma shows that, drawing a sample of polynomial size according to $\mathbf{m}_r$, we get, with high probability, all examples of low conditional Kolmogorov complexity (see [4]).

Lemma 3.2 *Drawing according to $\mathbf{m}_r$ a sample S of size $n^d(d \ln n + \ln 1/\delta)$, with probability greater than $1 - \delta$, S includes all the examples in $S_d^r(n)$.*

Finally, we show the main result.

Theorem 3.3 *Let $\mathcal{R}$ be a representation class learnable by membership and equivalence queries, then $\mathcal{R}$ is also learnable in the PACS model.*

Proof Let $r \in \mathcal{R}$ the target concept and let L be a membership and equivalence queries algorithm for $\mathcal{R}$. Let S be a large enough polynomial sample drawn according to $\mathbf{m}_r$. The size of S only depends on L and some constants. According to Lemmas 3.1 and 3.2, with high probability, S includes, as labelled examples, all the membership queries made by L in the computation on r with teacher T_{min}, and all the lexicographically smallest counterexamples to its wrong hypothesis. Now, we simulate the computation of L answering its membership queries according to its labels in S, and solving its equivalence queries finding the lexicographically smallest counterexample in S. The simulation ends when no counterexample is found to an equivalence query or a membership query can not be answered. So, with high probability the simulation outputs an hypothesis equivalent to the target. $\square$

It is important to note that we do not use the full power of query learnability in Theorem 3.3. First, we only need the existence of a polynomial time learner when T_{min} is the teacher, i.e. a learner that may not work with other teachers. And second, we can also apply the same proof to other teacher T and type of query Q, if there is a polynomial time algorithm that decides which is the answer given by T to an input to Q from the input to Q and a sample that contains the answer that T would give. So Theorem 3.3 can be generalized to learning algorithms that work with teachers and queries that satisfy the previous condition. We do not formalize this issue to keep the understandability of the paper and because it is straightforward.

Also, from Theorem 3.3 immediately follows that the class of DFAs, that Angluin showed learnable by membership and equivalence queries in [1], is learnable in the PACS model. This result was shown in a recent paper ([6]) by an argumentation specific to DFA.

4 PACS vs. simple-PAC

In this section we study the relationships between PACS and simple-PAC learning models. Assuming some properties in the learning algorithms we can show that both models are strongly correlated. Moreover, these properties are satisfied for all known learning algorithms in these models.

We start relating distributions $\mathbf{m}_\lambda$ and $\mathbf{m}_r$. Let $\mathcal{R}$ be a representation class and $r \in \mathcal{R}$ be a concept.

Lemma 4.1 *There exist constants $k_1 \neq 0$ and k_2, independent of x and r such that*

1. $\mathbf{m}_r(x)2^{-K(r)}k_1 \leq \mathbf{m}_\lambda(x)$.
2. $\mathbf{m}_r(x)2^{K(r)}k_2 \geq \mathbf{m}_\lambda(x)$.
3. *Let S be a sample of size $\frac{2^{K(r)}n^d}{k_1}(d \ln n + \ln 1/\delta)$ drawn according to $\mathbf{m}_\lambda$. The probability of the event $S \supseteq S^r_d(n)$ is at least $1-\delta$.*

Proof We use the following well known inequalities that relate $K(x)$ and $K(x|r)$ (see [5]):
(a) $K(x|r) \leq K(x) + O(1)$.
(b) $K(x) \leq K(x|r) + K(r) + O(1)$.

To prove *(i)* we use (b) and get

$$\mu_\lambda \mu_r 2^{-K(x)} \geq \mu_\lambda \mu_r 2^{-K(x|r)-K(r)-O(1)}$$

where μ_λ and μ_r are the constants such that $\mathbf{m}_\lambda(x) = \mu_\lambda 2^{-K(x)}$ and $\mathbf{m}_r(x) = \mu_r 2^{-K(x|r)}$. Rewriting that last inequality in terms of probabilities we obtain,

$$\mu_r \mathbf{m}_\lambda(x) \geq \mu_\lambda \mathbf{m}_r(x) 2^{-K(r)-O(1)}$$

To finish the proof it is enough to show that $\mu_\lambda/\mu_r \geq c$, for some constant c. To see this, note that using inequality (a)

$$1 = \mu_\lambda \sum_x 2^{-K(x)} = \mu_r \sum_x 2^{-K(x|r)} \geq \mu_r 2^{-O(1)} \sum_x 2^{-K(x)},$$

that shows that μ_λ/μ_r is bounded below by a constant greater than zero.

To prove *(ii)* we use (a) and get

$$\mu_\lambda \mu_r 2^{-K(x|r)} \geq \mu_\lambda \mu_r 2^{-K(x)-O(1)}$$

therefore,

$$\mu_\lambda \mathbf{m}_r(x) \geq \mu_r \mathbf{m}_\lambda(x) 2^{-O(1)}$$

now it is enough to show that μ_λ/μ_r is $O(2^{K(r)})$, which can be done using inequality (b) as follows:

$$1 = \mu_r \sum_x 2^{-K(x|r)} = \mu_\lambda \sum_x 2^{-K(x)} \geq \mu_\lambda 2^{-O(1)-K(r)} \sum_x 2^{-K(x|r)},$$

that shows that μ_λ/μ_r is $O(2^{K(r)})$.

To show *(iii)* note that *(i)* implies that the $\mathbf{m}_\lambda$ probability of a word x in $S_d^r(n)$ is at least $k_1 2^{-K(r)} n^{-d}$. Now using an standard analysis as in Lemma 3.2 the statement follows. □

Let $\mathcal{R}$ be a representation class that is PACS learnable, and A be a learning-from-examples algorithm for $\mathcal{R}$. We say that A has the property P if A satisfies the following: There exist a constant d and a polynomial p such that for any $r \in \mathcal{R}$ given a sample S of size at least $p(\frac{1}{\epsilon}, n, l(r))$, if $S \supseteq S_d^r(n)$ then A with input S outputs a hypothesis h such that $\mathbf{m}_r(h \triangle r) < \epsilon$.

For a representation class $\mathcal{R}$ we define as simple($\mathcal{R}$) the set $\{r \in \mathcal{R} : K(r) \leq c \log(l(r))\}$ for an arbitrary c that we fix for the rest of the paper, as Li and Vitányi do in [4].

Theorem 4.2 *Let $\mathcal{R}$ be a PACS learnable representation class that has a learning algorithm with property P. Then, simple($\mathcal{R}$) is simple-PAC learnable.*

Proof Let A be a PACS learning algorithm for $\mathcal{R}$ with property P, i.e., for some constant d and polynomial p, given a sample S of size $p(\frac{1}{\epsilon'}, n, l(r))$ such that $S \supseteq S_d^r(n)$, then A outputs a hypothesis h such that $\mathbf{m}_r(h \triangle r) < \epsilon'$.

Let r be a concept with $K(r) \leq c \log(l(r))$. According to the last statement of Lemma 4.1, given a $\mathbf{m}_\lambda$ sample S' of size

$$\max(p(\frac{1}{\epsilon'}, n, l(r)), \frac{l(r)^c n^d}{k_1}(d \ln n + \ln 1/\delta)),$$

with probability greater than $1 - \delta$, S' includes all words in $S_d^r(n)$. Therefore , A outputs a hypothesis h such that $\mathbf{m}_r(h \triangle r) < \epsilon'$ with $\mathbf{m}_\lambda$ propability greater than $1 - \delta$. Using statement *(ii)* of Lemma 4.1

$$\mathbf{m}_\lambda(h \triangle r) \leq \mathbf{m}_r(h \triangle r) l(r)^c k_2 < \epsilon' l(r)^c k_2,$$

which shows that an appropiate choice of ϵ', namely ϵ' such that $\epsilon' l(r)^c k_2 < \epsilon$, suffices for the result. □

It might seem that property P is a strong requirement; but all known ([3] and [6]) PACS learning algorithms satisfy it. In particular, all algorithms obtained by applying Theorem 3.3 trivially fulfill property P.

An interesting consequence of Theorems 3.3 and 4.2 is that the simple concepts of a query learnable concept class, are simple-PAC learnable in the sense of Li and Vitányi. Formally,

Corollary 4.3 *If $\mathcal{R}$ is learnable by a membership and equivalence query algorithm, then simple($\mathcal{R}$) is simple PAC learnable.*

For the converse of Theorem 4.2 we cannot give an absolute answer, but we note that, if simple($\mathcal{R}$) has a simple-PAC learning algorithm that works independently of the universal prefix Turing machine used to define the underlying Kolmogorov complexity, then the same algorithm is a PACS learning algorithm for the whole class $\mathcal{R}$. Again, all known simple-PAC algorithms satisfy this requirement. As far as we know, there is no counterexample for the equivalence between PACS and simple-PAC of simple concepts.

We note that the previous condition is not trivially satisfied. If we use two different universal prefix Turing machines U_1 and U_2, the corresponding universal distributions $\mathbf{m}_{U_1}$ and $\mathbf{m}_{U_2}$ are equal up to a multiplicative constant. This does not seem to be enough to guarantee that a simple-PAC learning algorithm that works with $\mathbf{m}_{U_1}$ has to work also with $\mathbf{m}_{U_2}$.

Observe that Li and Vitányi showed in [4] that superclasses of simple(DNF) and simple(k-reversible Automata) are simple-PAC learnable. As a consequence of the previous discussion DNF and k-reversible Automata are PACS learnable classes, which subsumes [3].

5 Conclusions and Further Work

We have studied a new variation of distribution dependent PAC learning, the PACS model. We have shown that all known positive results in this model could be derived from previous results in query and simple-PAC learning models, using our general technique.

The conditions that appear in the results of Section 4 –property P and independence of the universal prefix Turing machine– open a line of research. Is property P necessary in Theorem 4.2? Are all simple-PAC learning algorithms independent of the universal prefix Turing machine used to define the underlying Kolmogorov complexity?

Another interesting question and maybe more fundamental is if simple-PAC learnability is independent of the universal prefix Turing machine of reference. Note that this question differs from the last in the previous paragraph in the sense that we permit different algorithms for different Turing machines.

References

[1] D. Angluin. "Learning regular sets from queries and counterexamples". Information and Computation, 75:87–106, November 1987.

[2] D. Angluin. "Queries and concept learning". Machine Learning, 2(4):319–342, 1988.

[3] C. Denis, F. D'Halluin and R. Gilleron. "Pac learning with simple examples". In *Proc. of the* 13^{th} *Annual Syposium on the Theoretical Aspects of Computer Science*, pages 231–242. Lecture Notes in Computer Science, 1046. Springer, 1996.

[4] M. Li and P. Vitányi. "Learning simple concepts under simple distributions". SIAM Journal of Computing 20, pages 911–935, 1991.

[5] M. Li and P. Vitányi. "An introduction to Kolmogorov complexity and its applications". Springer-Verlag, 1993.

[6] R. Parekh and V. Honavar. "Learning dfa from simple examples". In *Proc. of the* 8^{th} *International Workshop on Algorithmic Learning Theory*, pages 116–131. Lecture Notes in Artificial Intelligence, 1361. Springer, 1997.

[7] L. Valiant. "A theory of the learnable". *Comm. ACM*,27:1134–1142, 1984.

[8] O. Watanabe. "A framework for polynomial time query learnability". *Math. Systems Theory*, 27:211–229, 1992.