Computational Learning Theory
FoPPS Logic and Learning School

Lecturer: James Worrell

1 Overview

These notes give a short introduction to the *Probably Approximately Correct (PAC)* learning model. Within this model one can analyse a variety of popular approaches to classification problems, including neural nets, support-vector machines, and boosting. We present a simple formulation of the PAC model in which the goal of the learner is to identify an unknown target function $c : \mathcal{X} \to \{0, 1\}$ drawn from a known class of functions (this is often called the realisable setting). More precisely, based on the values taken by c on a sufficiently large sample drawn from a fixed but arbitrary distribution D on $\mathcal{X}$, the learner must infer a hypothesis function h that is "approximately equivalent" to c in the sense that c and h agree with high probability on fresh samples from D. The notion of PAC learning is distribution independent: we don't say anything in these notes about learning under specific distributions.

While the PAC model is formulated in terms of *prediction* we describe the recently discovered result that PAC learnability can equivalently be characterised in terms of *sample compression*. We also give a classical combinatorial measure of the sample complexity of PAC learning in terms of VC dimension, focussing particularly on the VC dimension of concept classes defined within predicate logic. Finally we turn our attention to the computational complexity of learning. We show that certain concept classes are not *efficiently* PAC learnable under standard cryptographic assumptions, and, motivated by this, we introduce an extension of the PAC model with membership queries.

The notes are arranged as follows.

- In Section 2 we introduce the PAC model, considering only the amount of information required for learning and ignoring computational questions.
- In Section 3 we define the notions of VC dimension and growth function of a concept class. We calculate the VC dimension of various geometric concept classes and uniformly definable families of sets in predicate logic. We obtain VC dimension bounds for certain neural net architectures.
- In Section 4 we prove the fundamental result that a concept class has finite VC dimension if and only if it is PAC learnable.
- In Section 5 we introduce the notion of a sample compression scheme, which is a natural abstraction of many different learning procedures. We show that a concept class is PAC learnable if and only if it has a sample compression scheme (the "only if" direction here is a recent result, due to Moran and Yehudayoff [14], that settled a longstanding conjecture of Littlestone and Warmuth [10]).
- In Section 6 we consider the computational complexity of learning. We observe that certain concept classes, including finite automata, that can be learned with polynomial many samples can nevertheless not be learned in polynomial time under standard cryptographic assumptions.
- In Section 7, motivated by the above-mentioned cryptographic hardness results, we extend the PAC model with membership queries and give a polynomial-time algorithm for learning learning weighted automata in this setting.

2 The PAC Model

2.1 Definition of PAC Learnability

A learning problem in the PAC model is specified by an *input space* $\mathcal{X}$ and *concept class* $\mathcal{C}$, where $\mathcal{C}$ is a family of functions from $\mathcal{X}$ to $\{0,1\}$ (or, equivalently, a family of subsets of $\mathcal{X}$). An instance of such a learning problem in determined by a *target concept* $c \in \mathcal{C}$ and a fixed but unknown distribution D on $\mathcal{X}$. The output of the learner is a hypothesis $h \in \{0,1\}^{\mathcal{X}}$. This output is generated based on a finite sample S drawn i.i.d. from D and labelled by c. We define the *generalisation error* of h to be

$$\mathrm{err}(h) \stackrel{\mathrm{def}}{=} \Pr_{x \sim D}\big(h(x) \neq c(x)\big).$$

The goal of a learner is that h be *probably approximately correct*, where the term "approximately" is quantified through an *accuracy parameter* ε and the term "probably" is quantified through a *confidence parameter* δ. Specifically we require that that $\mathrm{err}(h) \leq \varepsilon$ with probability at least $1-\delta$. The probability here is with respect to the random sample S—intuitively we cannot rule out the unlucky event that the learner draws an unrepresentative training set and is unable to infer a good approximation of c.

The formal definition of a PAC learnable concept class $\mathcal{C}$ is as follows. Define $L_{\mathcal{C}}(m)$ to consist of the collection of labelled samples, i.e., pairs $(S, c|_S)$ where $S \in \mathcal{X}^m$ and $c \in \mathcal{C}$. We say that $\mathcal{C}$ is *PAC learnable with sample complexity m, accuracy ε, and confidence δ* if there is a *learning map* $H : L_{\mathcal{C}}(m) \to \{0,1\}^{\mathcal{X}}$ such that for any target concept $c \in \mathcal{C}$ and distribution D on $\mathcal{X}$, we have that

$$\Pr_{S \sim D^m}\big(\mathrm{err}(H(S, c|_S)) \leq \varepsilon\big) \geq 1 - \delta\,.$$

We furthermore say that H is a *proper learning map* if the range of H is included in the class $\mathcal{C}$.

The above definition of PAC learnability abstracts from the representation and computability of the learning map: for now our only concern is the the number of samples required for learning.

2.2 Learning Rectangles in the Plane

We illustrate the notion of a PAC learnable class with the following classic example. Let the input space $\mathcal{X}$ be $\mathbb{R}^2$ and consider the concept class $\mathcal{C}$ of all rectangles in the plane with sides parallel to the coordinate axes. Fix a distribution D on $\mathbb{R}^2$ and target concept $R \subseteq \mathbb{R}^2$. We show that by drawing a suitably large finite sample S from D a learner can with probability at least $1-\delta$ output a rectangle R' such that

$$\mathrm{err}(R') = \Pr_{x \sim D}(x \in (R \setminus R') \cup (R' \setminus R)) \leq \varepsilon\,.$$

This goal can be realised very simply. Given a sample S, define the hypothesis rectangle R' to be the smallest rectangle that is consistent with the sample, i.e., the smallest rectangle that includes all the positive examples. Notice that $R' \subseteq R$ by construction (see Figure 1) and therefore R' necessarily excludes all negative examples.

We bound the error of the hypothesis as follows. Given $E \subseteq \mathbb{R}^2$, write $\Pr(E)$ for $\Pr_{x \sim D}(x \in E)$. Now if $\Pr(R) \leq \varepsilon$ then clearly $\Pr(R \setminus R') \leq \varepsilon$. Otherwise define four "border rectangles" $E_1, E_2, E_3, E_4 \subseteq R$ (shown in Figure 1) such that $\Pr(E_1) = \Pr(E_2) = \Pr(E_3) = \Pr(E_4) = \varepsilon/4$ and $E_1 \cup E_2 \cup E_3 \cup E_4$ contains the boundary of R.[1]

If each border region E_i contains at least one sample point then the hypothesis rectangle R' is such that $R \setminus R' \subseteq E_1 \cup E_2 \cup E_3 \cup E_4$. In this case,

$$\begin{aligned}\Pr(R \setminus R') &\leq \Pr(E_1 \cup E_2 \cup E_3 \cup E_4)\\ &\leq \sum_{i=1}^{4} \Pr(E_i) \leq \varepsilon\,.\end{aligned}$$

[1] In general it may not be possible select the E_i with measure *exactly* $\varepsilon/4$. The construction can be appropriately generalised, but we will ignore this distracting possibility.

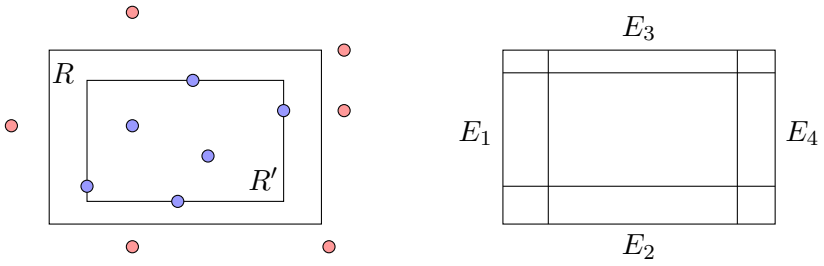

Figure 1: Concept and hypothesis rectangles; border regions.

It remains to give a lower bound on the sample size m ensuring that with probability at least $1-\delta$ the sample contains a point in each border region E_i. Now the probability that a given sample point misses E_1 is at most $1-\varepsilon/4$. Since the samples are drawn independently, the probability that all samples miss E_1 is at most

$$(1-\varepsilon/4)^m \leq e^{-\varepsilon m/4},$$

using the inequality $1+x \leq e^x$. The same reasoning applies to each E_i. By a union bound, the probability that the sample misses some border region E_i is at most $4e^{-\varepsilon m/4}$. This quantity is at most δ if $m \geq (4/\varepsilon)\log(4/\delta)$. Thus we can achieve the desired error and confidence bounds using a sample of size linear in $1/\varepsilon$ and logarithmic in $1/\delta$.

This concludes our first example of a PAC learnable class. In Section 3 we give a sufficient and necessary criterion for PAC learnability of a concept class in terms of VC dimension. From this characterisation it follows, e.g., that the concept class of all convex polygons in the plane is not PAC learnable.

3 VC Dimension

3.1 Definition

Let $\mathcal{C}$ be a concept class on input set $\mathcal{X}$. We say that $S \subseteq \mathcal{X}$ *shattered* by $\mathcal{C}$ if every function from S to $\{0,1\}$ arises as the restriction of some $c \in \mathcal{C}$. The *VC dimension* of $\mathcal{C}$ is defined by

$$\mathrm{VC}(\mathcal{C}) = \sup\left\{|S| : S \text{ a finite subset of } \mathcal{X} \text{ that is shattered by } \mathcal{C}\right\}.$$

Figure 2 gives the VC dimension of some geometric concept classes. Notice that in all but the last example the VC dimension corresponds to the number of real-valued parameters that define a concept. We will justify the VC-dimension bounds in Figure 2 in the lecture. Here we give two cases by way of example.

Example: half-spaces in $\mathbb{R}^n$. We argue that the VC dimension of the class of half-spaces in $\mathbb{R}^n$ is at most $n+1$, i.e., we prove that no set of $n+2$ points in $\mathbb{R}^n$ is shattered. Indeed given such a set S, by Theorem 1 we can partition S into two disjoint sets S_1 and S_2 whose convex hulls meet. But then there is no half space that includes all points in S_1 and excludes all points in S_2.

Theorem 1 (Radon's Theorem)**.** *Any set of $n+2$ points $S \subseteq \mathbb{R}^n$ can be partitioned into two subsets S_1 and S_2 such that the convex hulls of S_1 and S_2 intersect.*

Example: trigonometric functions. We argue that the class $\{x \mapsto \mathrm{sgn}(\sin \alpha x) : \alpha \in \mathbb{R}\}$ has infinite VC dimension. To this end, consider $S = \{x_1, \ldots, x_m\} \subseteq \mathbb{R}$, where $x_i = 2^{-i}$, together

$\mathcal{X}$	$\mathcal{C}$	$\mathrm{VC}(\mathcal{C})$
$\mathbb{R}^2$	axis-aligned rectangles	4
$\mathbb{R}^2$	convex k-gons	$2k+1$
$\mathbb{R}^n$	half spaces	$n+1$
$\mathbb{R}$	$\{x \mapsto \operatorname{sgn}(\sin \alpha x) : \alpha \in \mathbb{R}\}$	∞

Figure 2: VC dimension of geometric concept classes

with an arbitrary labelling $f : S \to \{-1, +1\}$. Define $\alpha := \frac{\pi}{2}\left(1 + \sum_{i=1}^{m} 2^i(1 - f(x_i))\right)$. Then for $k = 1, \ldots, m$ we have

$$\begin{aligned} \alpha x_k \bmod 2\pi &= \alpha 2^{-k} \bmod 2\pi \\ &= \frac{\pi}{2}\left(2^{-k} + \sum_{i=1}^{k-1}(1 - f(x_i))2^{i-k}\right) + \frac{\pi}{2}(1 - f(x_k)) \\ &= c\pi + \frac{\pi}{2}(1 - f(x_k))\,, \end{aligned}$$

where $c \in (0, 1)$. It is thus clear that $\operatorname{sgn}(\sin \alpha x_k) = f(x_k)$ for $k = 1, \ldots, m$.

Given a concept class $\mathcal{C}$ on an input space $\mathcal{X}$, the *dual concept class* $\mathcal{C}^* \subseteq \{0, 1\}^{\mathcal{C}}$ comprises the set of functions $f_x : \mathcal{C} \to \{0, 1\}$, $x \in \mathcal{X}$, such that $f_x(c) = c(x)$ for all $c \in \mathcal{C}$.

Proposition 2. $\mathrm{VC}(\mathcal{C}) \leq 2^{\mathrm{VC}(\mathcal{C}^*)}$.

Proof. Suppose $\{c_1, \ldots, c_{2^n}\} \subseteq \mathcal{C}$ is shattered by $\mathcal{C}^*$ for some $n \in \mathbb{N}$. Then for each $i \in \{1, \ldots, n\}$ there exists $x_i \in \mathcal{X}$ such that $f_{x_i}(c_j) = 1$ if and only if the i-th bit of j is 1. But then $\{c_1, \ldots, c_{2^n}\}$ shatters $\{x_1, \ldots, x_n\}$. □

3.2 The Growth Function

In this section we define the *growth function* of a concept class and show that VC dimension can be used to obtain bounds on the growth function.

Consider a concept class $\mathcal{C}$ on input set $\mathcal{X}$. Given a finite sample $S \subseteq \mathcal{X}$, define

$$\Pi_{\mathcal{C}}(S) = \{c|_S : c \in \mathcal{C}\}\,.$$

The *growth function* of $\mathcal{C}$ is defined as

$$\Pi_{\mathcal{C}}(m) = \max_{S:|S|=m} |\Pi_{\mathcal{C}}(S)|.$$

Thus $\Pi_{\mathcal{C}}(m)$ gives the maximum number of labellings induced by $\mathcal{C}$ on a set of cardinality m.

Example: open intervals. By way of example, we calculate the growth function in case the hypothesis set $\mathcal{C}$ is the class of open sub-intervals of $\mathbb{R}$. Given a set S of m distinct points in $\mathbb{R}$, the subsets of S of the form $c \cap S$ for $c \in \mathcal{C}$ are precisely the subsets of S consisting of contiguous elements. Such subsets are determined by their (zero, one, or two) endpoints, and hence $\Pi_H(m) = \binom{m}{2} + \binom{m}{1} + \binom{m}{0} = O(m^2)$.

The following lemma gives an upper bound on the growth function in terms of VC dimension. It implies that either $\Pi_C(m)$ is polynomially bounded or $\Pi_{\mathcal{C}}(m) = 2^m$ for all m.

Lemma 3 (Sauer [15], Shelah [17])**.** *Let $\mathcal{C}$ be a hypothesis set with finite VC dimension d. Then for all $m \geq d$,*

$$\Pi_{\mathcal{C}}(m) \leq \sum_{k=0}^{d} \binom{m}{k} \leq \left(\frac{em}{d}\right)^d.$$

Considering $\mathcal{X} = \mathbb{N}$ and $\mathcal{C}$ the collection of subsets of $\mathbb{N}$ of cardinality at most d, one sees that the upper bound in Lemma 3 is tight.

3.3 Concept Classes from Predicate Logic

Let σ be a signature of predicate logic. For a σ-formula $\varphi(x_1, \ldots, x_m, y_1, \ldots, y_n)$, σ-structure $\mathcal{A}$, and elements $b_1, \ldots, b_n \in A$, we write

$$\varphi(\mathcal{A}, b_1, \ldots, b_n) := \{(a_1, \ldots, a_m) \in A^m : \mathcal{A} \models \varphi(a_1, \ldots, a_m, b_1, \ldots, b_n)\}\,.$$

We call $\varphi(\mathcal{A}, b_1, \ldots, b_n)$ the set defined by φ with parameters $b_1, \ldots, b_n$. Then the definable family of sets

$$\mathcal{C}(\varphi, \mathcal{A}) = \{\varphi(\mathcal{A}, b_1, \ldots, b_n) : b_1 \ldots, b_n \in A\}$$

specifies a concept class on the input space $\mathcal{X} = A^m$. We write $\mathrm{VC}(\varphi, \mathcal{A})$ for $\mathrm{VC}(\mathcal{C}(\varphi, \mathcal{A}))$.

In this section we show that $\mathrm{VC}(\varphi, \mathcal{A})$ is finite if $\mathcal{A} = (\mathbb{R}, 0, 1, +, \times)$ is the field of real numbers and, more generally, when $\mathcal{A}$ is an o-minimal structure. As we remark later, these results can be used to bound the VC dimension of fixed neural-net architectures with polynomial and sigmoidal activation functions.

Lemma 4 (Shelah [16])**.** *Let K be a class of structures such that for every formula $\varphi(x, y_1, \ldots, y_n)$ the set $\{\mathrm{VC}(\varphi, \mathcal{A}) : \mathcal{A} \in K\}$ is bounded. Then it also holds that $\{\mathrm{VC}(\varphi, \mathcal{A}) : \mathcal{A} \in K\}$ is bounded for every first-order formula $\varphi(x_1, \ldots, x_m, y_1, \ldots, y_n)$.*

The following result of semi-algebraic geometry is key to obtaining VC-dimension bounds of uniformly defined families over the reals. The result is an adaptation by Goldberg and Jerrum [4] of a result of Warren [19].

Theorem 5. *Let $P_1, \ldots, P_\ell$ be a set of polynomials of degree at most d in k real variables with $\ell \geq k$. Then the number of realisable sign assignments to the P_i (either positive, negative, or zero) is at most $(8ed\ell/k)^k$.*

With Theorem 5 in hand we can show:

Proposition 6. *Let $\mathcal{A} = (\mathbb{R}, 0, 1, +, \times)$ and let $\varphi(x_1, \ldots, x_n, y_1, \ldots, y_k)$ be a Boolean combination of s polynomial equalities and inequalities, with each polynomial mentioned in φ having degree at most d. Then $\mathrm{VC}(\varphi, \mathcal{A})$ is at most $2k \log(8eds)$.*

Proof. Suppose that $S \subseteq \mathbb{R}^n$ is a set of cardinality $m \in \mathbb{N}$ that is shattered by $\mathcal{C}$. For each $\boldsymbol{a} \in S$ the formula $\varphi(\boldsymbol{a}, \cdot)$ is defined by a Boolean combination of at most s equalities and inequalities involving polynomials of degree at most d. Consider the collection $\mathcal{P}$ of polynomials that appear in some formula $\varphi(\boldsymbol{a}, \cdot)$, $\boldsymbol{a} \in S$. The behaviour of $\varphi(\cdot, \boldsymbol{b})$ on S is determined by the signs of the polynomials in $\mathcal{P}$ when evaluated on $\boldsymbol{b}$. Since $|\mathcal{P}| \leq ms$, by Theorem 5 the number of realisable sign assignments of the polynomials in $\mathcal{P}$ is at most $\left(\frac{8edms}{k}\right)^k$. Since S is shattered we must have $2^m \leq \left(\frac{8edms}{k}\right)^k$ and, taking logarithms,

$$m \leq k \log(8eds) + k \log(m/k)\,.$$

We now consider two cases: if $m/k \leq 8eds$ then the right-hand term in the above inequality is at most $2k \log(8eds)$, while if $m/k \geq 8eds$ then we have $m \leq 2k \log(m/k)$, which entails the result. □

Recall that an ordered structure $\mathcal{A}$ is *o-minimal* if every definable subset of A is a finite union of intervals.

Proposition 7. *If $\mathcal{A}$ is an o-minimal structure then $\mathrm{VC}(\varphi, \mathcal{A}) < \infty$ for every first-order formula φ.*

Proof. By Theorem 4 it suffices to prove that $\mathrm{VC}(\varphi, \mathcal{A}) < \infty)$ for every formula $\varphi(x_1, y_1, \ldots, y_n)$. But it is shown in [11] that for such a formula φ there is a bound on the number intervals comprising $\varphi(\mathcal{A}, b_1, \ldots, b_n)$ that is independent of $b_1, \ldots, b_n \in A$. Since a set of $2k+1$ distinct elements cannot be shattered by the collection of all subsets of A that are comprised of unions of at most k intervals, the proof is complete. □

3.4 Uniform Bounds on VC Dimension over Classes of Structures

In this section we fix a signature σ comprising unary predicate symbols and a single binary relation symbol E. For a given σ-formula φ and class K of σ-structures, we aim to give upper bounds on $\{\mathrm{VC}(\varphi, \mathcal{A}) : \mathcal{A} \in K\}$. The following simple example shows that no such bounds hold in case K is the class of all finite graphs:

Example. Consider the formula $\varphi := E(x, y)$ and let $\mathcal{G}_n = (U, V, E)$ be a bipartite graph with sets of vertices $U = \{1, \ldots, n\}$ and $V = 2^{\{1,\ldots,n\}}$ and edges $E = \{(i, \alpha) : i \in U, \alpha \in V, i \in \alpha\}$. Clearly $\mathrm{VC}(\varphi, \mathcal{G}_n) \geq n$.

Say that a σ-structure $\mathcal{A}$ is a *tree* if the relation $E^{\mathcal{A}}$ is the graph of a partial function and there is a node r (the root) such that every other node has a unique path to r. We say that $\mathcal{A}$ is a *forest* if it is a disjoint union of trees.

Theorem 8 ([12])**.** *Let K be the class of σ-structures $\mathcal{A}$ such that the edge relation $E^{\mathcal{A}}$ is the graph of a function on A. Then for any formula $\varphi(x_1, \ldots, x_m, y_1, \ldots, y_n)$ the set $\{\mathrm{VC}(\varphi, \mathcal{A}) : \mathcal{A} \in K\}$ is bounded.*

Proof. By Theorem 4 it suffices to prove the result in the case $m = 1$.

Given $a, b \in A$ let $d(a, b)$ denote the distance of a and b in the underlying undirected graph of $\mathcal{A}$. Given $k, s \in \mathbb{N}$, denote by $\mathrm{tp}_k^{\mathcal{A}}(a_1, \ldots, a_s)$ the *k-type* of $a_1, \ldots, a_s \in A$, i.e. the set of all formulas $\psi(x_1, \ldots, x_s)$ of quantifier depth at most k that are satisfied by $a_1, \ldots, a_s$.

The idea of the proof is to consider two classes of sets that cannot be shattered by $\mathcal{C}(\varphi, \mathcal{A})$ and then to argue that any suitably large sample $S \subseteq A$ must contain a subset that is in one of the two classes of "unshatterable" sets. (Note in passing that the restriction on the class of structures K in the statement of the theorem only plays a role for the second class of unshatterable set.)

We first show that a set $S \subseteq A$ of cardinality $4n$ such that $\mathrm{tp}_k^{\mathcal{A}}(a) = \mathrm{tp}_k^{\mathcal{A}}(b)$ and $d(a, b) > 4k$ for all pairs of distinct elements $a, b \in A$ cannot be shattered. Indeed, consider such a set and fix a set of parameters $\boldsymbol{c} = (c_1, \ldots, c_n)$. For each c_i we have $d(c_i, a) < 2k$ for at most one element $a \in A$, hence there exists $S_1 \subseteq S$ of cardinality $3n$ such that $d(a, c_i) > 2k$ for all $a \in S_1$ and $i = 1, \ldots, n$. Now by locality of first-order logic we have that $\mathrm{tp}_k^{\mathcal{A}}(a\boldsymbol{c}) = \mathrm{tp}_k^{\mathcal{A}}(b\boldsymbol{c})$ for all $a, b \in S_1$ (see, e.g., [5, Corollary 25]). Hence the function $\varphi(\cdot, \boldsymbol{c})$ is constant on S_1. It follows that S cannot be shattered since there is no set of parameters $\boldsymbol{c}$ such that $\varphi(\cdot, \boldsymbol{c})$ maps exactly $2n$ elements of S to true.

Next we show that $\mathcal{C}(\varphi, \mathcal{A})$ cannot shatter $S = \{a_1, \ldots, a_{4n}\} \subseteq A$ such that there exist subtrees $\mathcal{T}_1, \ldots, \mathcal{T}_{4s}$ of $\mathcal{A}$ such that $a_i \in \mathcal{T}_i$, the respective roots of the $\mathcal{T}_i$ are all children of a common node, and $\mathrm{tp}_k^{\mathcal{T}_i}(a_i) = \mathrm{tp}_k^{\mathcal{T}_j}(a_j)$ for all $i, j \in \{1, \ldots, 4n\}$. Indeed, consider such a set S and fix a set of parameters $\boldsymbol{c} = (c_1, \ldots, c_n)$. For each c_i we have $c_i \in \mathcal{T}_j$ for at most one $j \in \{1, \ldots, 4n\}$ and hence without loss of generality we may suppose that $c_i \notin \mathcal{T}_j$ for all $i \in \{1, \ldots, n\}$ and $j \in \{1, \ldots, 3n\}$. It follows by a version of the composition method for forests (see, e.g., [5, Lemma 2.7]) that $\mathrm{tp}_k^{\mathcal{A}}(a_i\boldsymbol{c}) = \mathrm{tp}_k^{\mathcal{A}}(a_j\boldsymbol{c})$ for all $i, j \in \{1, \ldots, 3n\}$. Thus the function $\varphi(\cdot, \boldsymbol{c})$ is constant on $\{a_1, \ldots, a_{3n}\}$ and we conclude that there is no set of parameters $\boldsymbol{c}$ such that $\varphi(\cdot, \boldsymbol{c})$ maps exactly $2n$ elements of S to true.

It remains to argue that a sufficiently large sample $S \subseteq A$ contains one of the above two types of unshatterable set. We only give a very quick sketch. The argument has two ingredients. First by the Ramsey theorem, for all positive integers N_1 and N_2 there exists a positive integer M such that any set $S \subseteq A$ of cardinality M contains either $S_1 \subseteq S$ of cardinality N_1 such that $d(a, b) > 4k$ for all distinct

$a, b \in S_1$ or a subset $S_2 \subseteq S$ of cardinality N_2 such that $d(a,b) \leq 4k$ for all distinct $a, b \in S_2$. Using the fact that there are finitely many depth-k types in one variable one can show that for N_1 sufficiently large S_1 contains an unshatterable set of the first type and for N_2 sufficiently large S_2 contains an unshatterable set of the second type. □

A related result has been obtained by by Grohe and Turán [5], who showed that for φ an MSO formula over a relational signature σ and $w \in \mathbb{N}$, there is an upper bound on $\{\mathrm{VC}(\varphi, \mathcal{A}) : \mathcal{A} \in K\}$ for K the class of σ-structures of tree-width at most w. A class of graphs that has unbounded tree-width and is closed under subgraphs (minors?) contains arbitrarily large grids and hence, by the following example, cannot have bounded VC dimension.

Example: grid graphs. Let $\mathcal{G}_{m,n}$ denote the graph set of vertices $\{1, \ldots, m\} \times \{1, \ldots, n\}$ such that there is an edge from (i,j) to (i',j') if $|i - i'| + |j - j'| = 1$. Such a graph can be viewed as being embedded in the plane as a grid in an obvious manner. We describe a formula $\varphi(x,y)$ such that $\mathrm{VC}(\varphi, \mathcal{G}_{n,2^n}) \geq n$. The formula φ is such that $\mathcal{G}_{n,2^n} \models \varphi((i,1),(1,j))$ if the i-th bit in the binary expansion of j is 1. Then the concept class $\mathcal{C}(\varphi, \mathcal{G}_{n,2^n})$ shatters the set $\{(1,1), \ldots, (n,1)\}$. The definition of φ involves an existentially quantified monadic variable X such that $X(i,j)$ holds precisely when the i-th bit in the binary expansion of j is 1. (The formula φ uses the predicate X to simulate a binary counter: for $j = 1, \ldots, 2^n$ the bit vector $X(\cdot, j)$ gives the binary expansion of j.) The formula $\varphi(x,y)$ expresses that the vertical path starting at x meets the horizontal path starting at y at a vertex z such that $X(z)$ is true.

3.5 VC Dimension Bounds for Neural Nets

In this section we consider multilayered feedforward neural nets with a single output that use $\mathrm{sgn}(x)$ as activation function. Each neuron in such a network implements a linear threshold function.

Theorem 9. *Fix an architecture with n_0 inputs and ω parameters and write $\mathcal{C} \subseteq \{0,1\}^{\mathbb{R}^{n_0}}$ the for class of functions that can be implemented by instantiating the parameters. Then*

$$\mathrm{VC}(\mathcal{C}) \leq 2\omega \log_2(e\omega)\,.$$

Proof. Suppose that the network has depth d and n_i neurons in the i-th level for $i = 0, \ldots, d$. Let $\omega_{i,j}$ be the number of free parameters of the j-th neuron on level i and let $\mathcal{C}_{i,j} \subseteq \{0,1\}^{\mathbb{R}^{n_{i-1}}}$ be the class of functions that can be implemented by this neuron. Then the VC dimension of $\mathcal{C}_{i,j}$ is equal to $\omega_{i,j}$ and hence $\Pi_{\mathcal{C}_{i,j}}(m) \leq \left(\frac{em}{w_{i,j}}\right)^{\omega_{i,j}}$ by Sauer's Lemma. It follows that

$$\begin{aligned}
\Pi_{\mathcal{C}}(m) &\leq \prod_{i=1}^{d} \prod_{j=1}^{n_i} \Pi_{\mathcal{C}_{i,j}}(m) \\
&\leq \prod_{i=1}^{d} \prod_{j=1}^{n_i} \left(\frac{em}{\omega_{i,j}}\right)^{\omega_{i,j}} \\
&\leq (em)^{\omega} \quad (\text{since } \omega_{i,j} \geq 1)\,.
\end{aligned}$$

Now for any positive integer m we have that $\mathrm{VC}(\mathcal{C}) < m$ if $2^m > (em)^{\omega}$. This is satisfied for $m = 2\omega \log_2(e\omega)$ for all $\omega > 1$. Since the result holds trivially when $\omega = 1$ the proof is complete. □

4 VC Classes are PAC Learnable

In this section we show that concept classes with finite VC dimension are learnable with a sample bound that is linear in the VC dimension and polynomial in $1/\varepsilon$ and $1/\delta$. In fact we show that any function

that returns a consistent hypothesis given a "sufficiently large" sample is a PAC learning map. At this stage we don't say anything about the computational complexity of finding a consistent hypothesis.

Let $\mathcal{C}$ be a concept class on input set $\mathcal{X}$ and $c \in \mathcal{C}$ a given target concept. Given a finite set $S \subseteq \mathcal{X}$ the *empirical error* of $h \in \mathcal{C}$ on S is defined to be $\mathrm{err}_S(h) := \frac{1}{|S|}\sum_{x\in S} \mathbb{I}(h(x) = c(x))$. In this section we show, roughly speaking, that if $\mathcal{C}$ has finite VC dimension then there exists m such that for any distribution D on $\mathcal{X}$, with high probability a sample $S \sim D^m$ is such that $\mathrm{err}_S(h)$ is close to $\mathrm{err}(h)$ *for all* $h \in \mathcal{C}$. Moreover the sample size m is independent of the distribution D.

Let $S = \{x_1, \ldots, x_{2m}\}$ be an arbitrary list of elements of $\mathcal{X}$ for some positive integer m, and consider the following random experiment: for each index $i = 1, \ldots, m$, swap x_i with x_{i+m} with probability $1/2$ and otherwise do nothing. Now define $S_1 := \{x_1, \ldots, x_m\}$ and $S_2 := \{x_{m+1}, \ldots, x_{2m}\}$.

Proposition 10. *Given $\varepsilon > 0$, the probability that there exists $h \in \mathcal{C}$ such that $|\mathrm{err}_{S_1}(h) - \mathrm{err}_{S_2}(h)| > \varepsilon$ is at most $\Pi_{\mathcal{C}}(2m) 2\exp(\frac{-m\varepsilon^2}{2})$.*

Proof. Fix $h \in \mathcal{C}$. Consider independent random variables X_i, $i \in \{1, \ldots, m\}$, defined by

$$X_i := \begin{cases} -1 & \text{if } c(s_i) = h(s_i) \text{ and } c(s'_i) \neq h(s'_i) \\ +1 & \text{if } c(s_i) \neq h(s_i) \text{ and } c(s'_i) = h(s'_i) \\ 0 & \text{otherwise.} \end{cases}$$

Writing $X = \frac{1}{m}(X_1 + \cdots X_m)$ and applying Hoeffding's Inequality (see Section A), we have that

$$\Pr(|\mathrm{err}_{S_1}(h) - \mathrm{err}_{S_2}(h)| > \varepsilon) = \Pr(|S| > \varepsilon) \leq 2\exp\left(\frac{-m\varepsilon^2}{2}\right).$$

The proposition now follows by taking a union bound over the $\Pi_{\mathcal{C}}(2m)$ possibilities for $h|_S$. □

Theorem 11 (Uniform Convergence)**.** *Let $\mathcal{C}$ be a concept class on input set $\mathcal{X}$ that has VC dimension d. Then there is an absolute constant c_0 such that for any $\varepsilon, \delta > 0$, if*

$$m \geq \frac{c_0}{\varepsilon^2}\left(\log\frac{1}{\delta} + d\log\frac{d}{\varepsilon}\right) \tag{1}$$

then for any target concept $c : \mathcal{X} \to \{0, 1\}$ and distribution D on $\mathcal{X}$,

$$\Pr_{S\sim D^m}\left(\exists h \in \mathcal{C} : |\mathrm{err}_S(h) - \mathrm{err}(h)| > \varepsilon\right) \leq \delta.$$

Proof. Choose a sample $S = \{x_1, \ldots, x_{2m}\} \in \mathcal{X}^{2m}$ i.i.d. from distribution D. For each index $i \in \{1, \ldots, m\}$ swap x_i and x_{i+m} with probability $1/2$ and otherwise do nothing. Writing $S_1 := \{x_1, \ldots, x_m\}$ and $S_2 := \{x_{m+1}, \ldots, x_{2m}\}$, the resulting distribution on S_1 and S_2 is the same as if we were to draw two lists of m elements i.i.d. from D.

Let A be the event that there exists $h \in \mathcal{C}$ with $|\mathrm{err}_{S_1}(h) - \mathrm{err}(h)| > \varepsilon$. We show that $\Pr(A) \leq \delta$. To this end, consider the event B that there exists $h \in \mathcal{C}$ with $|\mathrm{err}_{S_1}(h) - \mathrm{err}_{S_2}(h)| > \varepsilon/2$ together with the random variable $\Pr(B \mid S_1)$—the probability of B conditioned on S_1. We argue that $\Pr(B \mid S_1) \geq \frac{1}{2}$ if S_1 corresponds to an outcome in event A, i.e., if there exists $h_1 \in H$ with $|\mathrm{err}_{S_1}(h_1) - err(h_1)| > \varepsilon$. Indeed, since S_1 and S_2 are independent, the distribution of S_2 conditioned on S_1 is identical to D^m. Hence by Hoeffding's Inequality and the bound (1) we have

$$\Pr(B \mid S_1) \geq \Pr(|\mathrm{err}_{S_2}(h_1) - \mathrm{err}(h_1)| \leq \varepsilon/2) \geq 1/2.$$

It follows that $\Pr(B) \geq \Pr(B \cap A) \geq 1/2\Pr(A)$ and hence $\Pr(A) \leq 2\Pr(B)$.

Thus to show that $\Pr(A) \leq \delta$ it suffices to prove that $\Pr(B) \leq \delta/2$. To this end, write $\Pr(B \mid S)$ for the probability of B conditioned on a particular outcome S of the initial sample of $2m$ elements from $\mathcal{X}$. Then we have

$$\Pr(B \mid S) \leq \Pi_{\mathcal{C}}(2m) 2 \exp\left(-\varepsilon^2 m/8\right) \leq \left(\frac{2em}{d}\right)^d 2 \exp\left(-\varepsilon^2 m/8\right) ,$$

where the first inequality follows from Proposition 10 and the second follows from Corollary 3. By simple algebra it can be shown that for c_0 sufficiently large, if m satisfies the bound (1) then the above expression for $\Pr(B \mid S)$ is at most $\delta/2$. Since this inequality holds irrespective of S, we have that $\Pr(B) \leq \delta/2$. □

Theorem 12. *A concept class that has finite VC dimension admits a proper PAC learning function.*

Proof. By Theorem 11 if $m \in \mathbb{N}$ satisfies (1) then any function that maps a labelled sample of size m to a consistent hypothesis is a PAC learning map. □

For later use we state the following immediate consequence of Proposition 16.

Corollary 13. *Let $\mathcal{C} \subseteq \{0,1\}^{\mathcal{X}}$ have VC dimension d. Let D be a distribution on $\mathcal{X}$ and $\varepsilon > 0$. Then there exists a multiset $S \subseteq \mathcal{X}$ of cardinality $O(d/\varepsilon^2)$ such that for all $h \in \mathcal{C}$,*

$$\left| \Pr_{x \sim D}(h(x) = 1) - \frac{1}{|S|} \sum_{x \in S} h(x) \right| < \varepsilon \,. \tag{2}$$

Proof. The result holds by applying Proposition 16 in the special case that $c : \mathcal{X} \to \{0,1\}$ is the constant zero function. Note that we seek the mere existence of a set S satisfying (2) while Proposition 16 gives bounds on the probability that a set S randomly chosen from D^m satisfies (2). □

The final result of this section shows that PAC learnable classes have finite VC dimension. In particular the result shows that the class of all convex polygons in the plane is not PAC learnable. The proof (omitted) is an application of the probabilistic method.

Theorem 14. *Let $\mathcal{C}$ be a concept class that has VC dimension at least d. Then for any learning algorithm there exists a target concept $c \in \mathcal{D}$ and distribution D on the input space $\mathcal{X}$ such that if the algorithm is given $d/2$ examples then the output hypothesis $h \in \mathcal{C}$ is such that*

$$\Pr(\mathrm{err}(h) > 1/8) > 1/8 \,.$$

5 Sample Compression Schemes

5.1 Definitions

The notion of a sample compression scheme captures a common property of many learning procedures. A sample compression scheme of size k for a concept class consists of a compression function and a reconstruction function. Given a finite set of examples, labelled by the target concept, the compression function returns a subset of examples of size at most k and some side information from a finite set. The reconstruction function uses the subset of examples to construct a hypothesis for the concept to be learned. The reconstructed hypothesis is required to predict the correct label for all examples in the original sample set.

Formally, a *sample compression scheme* for a concept class $\mathcal{C}$ consists of positive integer $k \in \mathbb{N}$, called the *kernel size*, and a finite information set I, together with:

- A *compression map*
$$\kappa : \bigcup_{m \in \mathbb{N}} L_{\mathcal{C}}(m) \to L_{\mathcal{C}}(k) \times I$$
mapping (S, f) to $((S', f'), \sigma)$, where $S' \subseteq S$ and $f' = f|_{S'}$.

- A *reconstruction map*
$$\rho : L_{\mathcal{C}}(k) \times I \to \{0,1\}^{\mathcal{X}},$$
such that $\rho(\kappa(S, f))|_S = f$ for all $(S, f) \in L_{\mathcal{C}}(m)$, $m \geq k$.

Example: rectangles in the plane. Our first example of a sample compression scheme concerns the concept class of axis-aligned rectangles, as described in Section 2.2. Here the compression function selects from a given sample a subset of at most four points—namely the points with the least and greatest x and y coordinates. The reconstruction function maps the four selected points to the smallest enclosing rectangle. Clearly this rectangle correctly predicts the label of all points in the original sample.

Example: intervals in the real line. Consider the concept class of all subsets of $\mathbb{R}$ that are formed of at most n intervals. We describe a sample compression scheme of kernel size $2n$. The compression function that scans a sample left-to-right, saves the first positive example, the first subsequent negative example, the first subsequent positive example, and so on. At most $2n$ points are saved by the compression map. The reconstruction map builds a union of left-closed right-open intervals whose endpoints are adjacent positive and negative examples from the compressed sample.

Example: support vector machines. This example assumes knowledge of the support vector machines algorithm for learning half-spaces in $\mathbb{R}^n$. Given a labelled sample, this algorithm returns a linear classifier of maximum margin. This algorithm can be seen as the composition of a compression map and a reconstruction map. The compression map selects from the sample a subset (of so-called support vectors) that determine a maximum margin classifier for the whole set. The reconstruction map builds this classifier from the support vectors.

Example: mistake-bounded algorithms. Our final example of a compression scheme assumes familiarity with notions from online learning. Consider a concept class $\mathcal{C} \subseteq \{0,1\}^{\mathcal{X}}$ that admits an online mistake-driven learning algorithm $\mathcal{A}$ with mistake bound k (e.g., the Winnow algorithm for learning monotone disjunctions.) We describe a compression scheme of kernel size k. Fix a linear order on $\mathcal{X}$, which we call the *default order*. The compression map sends a sample $(S, c|_S)$ to the subsequence of inputs for which algorithm $\mathcal{A}$ would predict an incorrect label if given S in the default order. In order to predict the label of some $x \in \mathcal{X}$ that does not lie in the compression set, the reconstruction map runs algorithm $\mathcal{A}$ on all the entries of the compression set that precede x in the default order, and then outputs the label for x that is predicted by $\mathcal{A}$.

Next we prove that every concept class that has a sample compression scheme is PAC learnable:

Theorem 15. *Let $\mathcal{C}$ be a concept class on input space $\mathcal{X}$ that has a sample compression scheme κ, ρ of kernel size k. Given $\varepsilon > 0$ and $\delta > 0$, let*
$$m \geq \max\left(\frac{2}{\varepsilon}\left(\log\left(\frac{2}{\delta}\right) + \log|I|\right), \frac{4k}{\varepsilon}\log\left(\frac{4k}{\varepsilon}\right) + 2k\right).$$
Then the function $H : L_{\mathcal{C}}(m) \to \{0,1\}^{\mathcal{X}}$ defined by $H(S, f) = \rho(\kappa(S, f))$ is a PAC learning map with generalisation error ε and failure probability δ.

Proof sketch. Let $c \in \mathcal{C}$ be the target concept and D a distribution on the input space $\mathcal{X}$. Consider a sample $S = \{x_1, \ldots, x_m\}$ of m points drawn i.i.d. from D. Given the labelled sample $(S, c|_S)$, the compression map determines a subset $T \subseteq \{1, \ldots, m\}$ of cardinality k and $\sigma \in I$ such that, writing $S' = \{x_i : i \in T\}$, the hypothesis returned by the learning map is $h_{T,\sigma} := \rho((S', c|_{S'}), \sigma)$. Note that $h_{T,\sigma}$ is a consistent hypothesis—it agrees with c on S. Intuitively our goal is to bound the probability (over the random sample S) that there exists a "bad" choice of T and σ, i.e., such that $h_{T,\sigma}$ is a consistent hypothesis for S and $\mathrm{err}(h_{T,\sigma}) > \varepsilon$.

Fix $\sigma \in I$ and $T \subseteq \{1, \ldots, m\}$ of cardinality at most k. For these fixed values of σ and T consider $h_{T,\sigma}$ (as defined above) as a random variable on the space of samples S. Since the sample points are independent, the probability that $\mathrm{err}(h_{T,\sigma}) > \varepsilon$ and $h_{T,\sigma}$ agrees with c on all samples x_i for $i \notin T$ is at most $(1-\varepsilon)^{m-k}$. Taking a union bound over all $|I|\binom{m}{k}$ choices of σ and T and doing some arithmetic (see [10] for details), we have that the probability that there exists some σ and T such that $\mathrm{err}(h_{T,\sigma}) > \varepsilon$ and $h_{T,\sigma}$ agrees with all c on S is at most δ. □

5.2 VC Classes have Sample Compression Schemes

Littlestone and Warmuth [10] conjectured that any concept class with finite VC dimension d has a compression scheme of kernel size d. Obtaining compression schemes for VC classes has been studied in the context of model theory. Johnson and Laskowski [7] showed that if $\mathcal{A}$ is an o-minimal structure then the concept class $\mathcal{C}(\varphi, \mathcal{A})$ has a compression scheme of kernel size equal to the number of parameters of φ. Livni and Simon [8] and Chernikov and Simon [3] show the existence of compression schemes in case $\mathcal{A}$ is an *NIP structure*, that is, a structure in which all definable families $\mathcal{C}(\varphi, \mathcal{A})$ have finite VC dimension. In this section we present a result of Moran and Yehudayoff [14] that a concept class of VC dimension d has a sample compression scheme of kernel size $2^{\mathrm{poly}(d)}$.

We will need two preliminary results. The first, Proposition 16, is a direct application of Theorem 13.

Proposition 16. *Let $\mathcal{C} \subseteq \{0,1\}^{\mathcal{X}}$ be a concept class such that the dual class $\mathcal{C}^* \subseteq \{0,1\}^{\mathcal{C}}$ has VC dimension d^*. Let D^* be a distribution over $\mathcal{C}$ and $\varepsilon > 0$. Then there is a multiset $\mathcal{F} \subseteq \mathcal{C}$ of size $O(d^*/\varepsilon^2)$ such that for all $x \in \mathcal{X}$,*

$$\left| \Pr_{h \sim D^*}(h(x) = 1) - \frac{1}{|\mathcal{F}|} \sum_{h \in \mathcal{F}} h(x) \right| \leq \varepsilon \,.$$

The second result we will need is Von Neumann's minimax theorem for zero-sum matrix games:

Theorem 17 (Minimax)**.** *Let $M \in \mathbb{R}^{m \times n}$ be a real matrix. Then*

$$\min_{p \in \Delta^m} \max_{q \in \Delta^n} p^\top M q = \max_{q \in \Delta^n} \min_{p \in \Delta^m} p^\top M q \,,$$

where Δ^ℓ denotes the set of distributions on $\{1, \ldots, \ell\}$ for $\ell \in \mathbb{N}$.

We now come to the main result of the section.

Theorem 18. *Let $\mathcal{C}$ be a concept class of VC dimension d and dual VC dimension d^*. Then $\mathcal{C}$ admits a sample compression scheme of kernel size $O(d \cdot d^*)$.*

The rest of this section is devoted to the proof of Theorem 18.

Since $\mathcal{C}$ has finite VC dimension there exists $s \in \mathbb{N}$ and a proper learning map $H : L_{\mathcal{C}}(s) \to \mathcal{C}$ such that for every target concept $c \in \mathcal{C}$ there exists a sample $T \subseteq \mathcal{X}$ of cardinality at most s such that $\mathrm{err}(H(T, c|_T)) \leq 1/3$.

Let $(S, c|_S)$ be a labelled sample to be compressed, where $S \subseteq \mathcal{X}$. Let $\mathcal{H} = \{H(T, c|_T) : T \subseteq S, |T| \leq s\}$ be the image of the learning map on subsets of S of cardinality at most s. We claim that there is finite subset $F \subseteq \mathcal{H}$ of cardinality $O(d^*)$ such that the label of each element of S can be determined by a majority vote among the concepts in $\mathcal{F}$.

Claim 19. *There are $k = O(d^*)$ functions $f_1 \ldots, f_k \in \mathcal{H}$ such that for all $x \in S$,*

$$|\{i : f_i(x) = c(x)\}| > k/2\,.$$

proof of claim. We will apply the minimax theorem to the $S \times \mathcal{H}$ matrix M such that $M_{x,f} = \mathbb{I}\{f(x) = c(x)\}$. By definition of the learning map H, for every distribution D on S there is $h \in \mathcal{H}$ such that such that

$$\Pr_{x \sim D}(h(x) = c(x)) \geq 2/3$$

Therefore, by the Minimax Theorem applied to the matrix M, there is a distribution D^* on $\mathcal{H}$ such that for every $x \in S$,

$$\Pr_{h \sim D^*}(h(x) = c(x)) \geq 2/3$$

By Proposition 16 applied to $\mathcal{H}$ and D^*, with $\varepsilon = 1/8$, there is a multiset $\{f_1, \ldots, f_k\} \subseteq \mathcal{H}$ of size $k = O(d^*)$ such that for every $x \in S$,

$$\frac{|\{i : f_i(x) = c(x)\}|}{k} \geq \Pr_{h \sim D^*}(h(x) = c(x)) - 1/8 > 1/2$$

This concludes the proof of the claim. □

Given the claim, we can proceed to describe a sample compression scheme for $\mathcal{C}$.

Compression. Let $Z_i \subseteq S$ be such $f_i = H(Z_i, c|_{Z_i})$ for $i = 1, \ldots, k$. We define the compression map κ by $\kappa(S, c|_S) = ((Z, c|_Z), \sigma)$ where $Z = \cup_{i=1}^k Z_i$ and the information σ identifies which elements of Z lie in each of the subsets $Z_1, \ldots, Z_k$. Note that $|Z| = O(s \cdot d^*) = O(d \cdot d^*)$.

Reconstruction. We define $h = \rho((Z, f), \sigma)$ as follows. Consider the sets $Z_1, \ldots, Z_k$ determined by Z and σ and define $f_i = H(Z_i, c|_{Z_i})$ for $i = 1, \ldots, k$. For every $x \in \mathcal{X}$, we define $h(x)$ to be the majority element in the list $f_1(x), \ldots, f_k(x)$.

6 Concepts that are Hard to Learn

In this section we turn our attention to the computational complexity of learning. Here it is natural to consider concept classes $\mathcal{C} = \{\mathcal{C}_n : n \in \mathbb{N}\}$ that are parameterised by the input size of a learning problem, e.g., for a given alphabet we might have $\mathcal{C}_n$ be the class of regular languages whose minimum automaton has n states. Under a widely believed cryptographic assumption, we'll show that there are classes such that $\mathrm{VC}(\mathcal{C}_n)$ grows polynomially in n (so there is no information theoretic barrier to efficient learning) and yet there is no polynomial-time PAC learning algorithm (due to the difficulty of predicting the value of a concept on a new input based on its behaviour on previously seen inputs). This negative result applies even in the case of *improper* learning, where the output hypothesis is allowed to come from any polynomially evaluable hypothesis class.

6.1 The Discrete Cube Root Assumption

In this section we describe the cryptographic assumption on which the hard-to-learn class is based.

Given primes p, q such that $p, q \equiv 2 \bmod 3$, write $N = pq$. Let $\mathbb{Z}_N^*$ denote the multiplicative group $\{a \in \mathbb{Z}_N : \gcd(a, N) = 1\}$. Then $|\mathbb{Z}_N^*| = \varphi(N) = (p-1)(q-1)$, where φ denotes Euler's totient function. By the choice of p and q we have that $\gcd(3, \varphi(N)) = 1$ and hence there exist $d, k \in \mathbb{N}$ such that $3d = 1 + k\varphi(N)$.

Consider the function $f_N : \mathbb{Z}_N^* \to \mathbb{Z}_N^*$, given by $f_N(x) = x^3 \bmod N$. This function is a bijection: indeed, defining $g_N : \mathbb{Z}_N^* \to \mathbb{Z}_N^*$ by $g_N(y) = y^d \bmod N$, then by Fermat's Little Theorem we have

$$f_N(g_N(x)) = g_N(f_N(x)) = x^{3d} = x^{1+k\varphi(N)} = x$$

for all $x \in \mathbb{Z}_N^*$. We naturally call g_N the discrete cube root function.

Given N and $x \in \mathbb{Z}_N^*$, suppose we want to compute $g_N(x)$. We can do this in polynomial time (in the bit length of N) if we know the prime factors p and q of N, since then we can compute $\varphi(N)$ and hence determine the exponent d. However it is a different matter if only N and x are known. In fact the following is a standard assumption in cryptography:

Definition 20 (Discrete Cube Root Assumption (DCRA))**.** *For any polynomial $P(\cdot)$, there does not exist any algorithm, A, that runs in time $P(n)$ and on input N and x, where N is the product of two random n-bit primes $p, q \equiv 2 \bmod 3$ and x is chosen randomly from $\mathbb{Z}_N^*$, outputs $g_N(x)$ with probability at least $1/P(n)$. The probability is over the random choices of p, q, x and any internal randomisation of A.*

6.2 A Learning Problem Based on DCRA

Let us try to phrase the question of finding the cube root of $x \in \mathbb{Z}_N^*$ as a learning question. Suppose that we have access to a sample, $(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)$, where $y_i = g_N(x_i)$ for $i = 1, \ldots, m$, and x_i are drawn uniformly at random from $\mathbb{Z}_N^*$. The learning question is, given such examples, can we obtain $h : \mathbb{Z}_N^* \to \mathbb{Z}_N^*$ such that for x drawn uniformly at random from $\mathbb{Z}_N^*$ it holds with probability at least $1 - \delta$ over the sample, that $\Pr(h(x) \neq g_N(x)) \leq \epsilon$?

Suppose there were a learning algorithm that computed such a function h in polynomial time in the bit size of N. We observe that this supposition would contradict DCRA. Indeed it is easy to generate a sample on which to train such an algorithm since, although finding the cube root is hard, finding the cube is easy. As g_N is a bijection, we can choose $y_i \in \mathbb{Z}_N^*$ uniformly at random, and then set $x_i = y_i^3 \bmod N$. Note that this implies that the distribution of x_i is uniform over $\mathbb{Z}_N^*$.

The above learning problem almost fits into our formulation of the PAC model, except that the output of the target function is not in $\{0, 1\}$. This can be easily fixed. We know that the output of g_N is some $2n$ bit string. Thus we consider $2n$ different target functions, $g_{N,i} : \mathbb{Z}_N^* \to \{0, 1\}$, for $i = 1, \ldots, 2n$, where $g_{N,i}$ outputs the i-th bit of the function g_N. If we could learn all the functions, $g_{N,i}$ to accuracy $\frac{\epsilon}{2n}$, then we could reconstruct g_N to accuracy ϵ. Thus, if learning g_N is hard, then at least one of the Boolean functions $g_{N,i}$ must also be hard to learn.

6.3 A Concept Class that is Hard to Learn

So far, we've established that if we choose random n bit primes p and q of the form $3k + 2$, there exists a Boolean function, $g_{N,i}$, such that if we get labelled examples from a *specific* distribution D over $2n$ bit strings, namely the uniform distribution over bit representations of elements in $\mathbb{Z}_N^*$, we cannot output a (polynomially evaluable) hypothesis h, such that $\Pr_{x \sim D}(g_{N,i}(x) \neq h(x)) \leq \frac{\varepsilon}{2n}$. If we can identify a parametrised concept class $\mathcal{C} = \{\mathcal{C}_n : n \in \mathbb{N}\}$ such that $g_{N,i} \in \mathcal{C}_{2n}$, then the class $\mathcal{C}$ cannot PAC-learnable in polynomial time.

Let us try and understand what such a concept class could be. First, we note that if d is known, there is a rather simple polynomial time algorithm to output $g_N(x)$. All we need to do is perform the operation $x^d \bmod N$. Naively computing x^d is not efficient as d may be as large as $\varphi(N)$, i.e., d may itself be $2n$ bits long. The first thing we need to ensure is that all operations are repeatedly performed modulo N; this way none of the representations get too large. The second is that we start by computing, $x \bmod N, x^2 \bmod N, x^4 \bmod N, x^8 \bmod N, \ldots, x^{2^{\lfloor \log \varphi(N) \rfloor}} \bmod N$, i.e., we compute $x^{2^i} \bmod N$ for $i = 0, 1 \ldots, \lfloor \log \varphi(N) \rfloor$. To obtain $x^d \bmod N$, we simply take the product of the terms $x^{2^i} \bmod N$ such that the i-th bit of d is 1. This shows that there exists a circuit of polynomial size that computes g_N where d is hard-wired into the circuit itself. In particular, this also implies that there exist polynomial-size circuits for $g_{N,i}$ for all $i = 1, \ldots, 2n$. This gives us the following result.

Theorem 21. *There exists a fixed polynomial $P(\cdot)$, such that class of circuits, $\mathcal{C}$, where $\mathcal{C}_n$ consists of circuits of size at most $P(n)$, is not PAC-learnable under the discrete cube root assumption.*

This result can be strengthened to show that for a fixed polynomial $P(\cdot)$, the class of circuits $\mathcal{C}$, where $\mathcal{C}_n$ consists of circuits of size at most $P(n)$ and depth $\log n$, is not PAC-learnable under the discrete cube-root assumption.

7 Learning Weighted Automata

7.1 Preliminaries

A weighted automaton over a field $\mathbb{K}$ is a tuple $\mathcal{A} = (n, \Sigma, \alpha, \{M(\sigma)\}_{\sigma \in \Sigma}, \eta)$ comprising the *dimension* $n \in \mathbb{N}$, *alphabet* Σ, *initial-state vector* $\alpha \in \mathbb{K}^n$, family of *transition matrices* $M(\sigma) \in \mathbb{K}^{n \times n}$, and *final-state vector* $\eta \in \mathbb{K}^n$. Extend M freely to Σ^* by writing $M(\sigma_1 \ldots \sigma_k) = M(\sigma_1) \cdots M(\sigma_k)$. Then $\mathcal{A}$ is said to *recognise* a formal power series $f : \Sigma^* \to \mathbb{K}$ if $f(w) = \alpha^\top M(w) \eta$ for all $w \in \Sigma^*$.

Write $e_i \in \mathbb{K}^n$ for the column vector with 1 in the i-th position and 0 in all other positions.

Define the *Hankel matrix* of a formal power series $f : \Sigma^* \to \mathbb{K}$ to be the infinite matrix F whose rows and columns are indexed by Σ^*, such that $F_{x,y} = f(xy)$ for $x, y \in \Sigma^*$. Recall that if f is recognised by a $\mathbb{K}$-weighted automaton $\mathcal{A}$ then the rank of its Hankel matrix is at most the number of states of $\mathcal{A}$.

7.2 The Algorithm

We describe an algorithm (from [1]) to exactly learn a weighted automaton computing a given function $f : \Sigma^* \to \mathbb{K}$ using membership and equivalence queries. In a membership query the learner asks for the value of f on a given word $w \in \Sigma^*$.

At each stage the algorithm maintains the following data:

- A set of n "rows" $X = \{x_1, \ldots, x_n\} \subseteq \Sigma^*$, where $x_1 = \varepsilon$.
- A set of n "columns" $Y = \{y_1, \ldots, y_n\} \subseteq \Sigma^*$, where $y_1 = \varepsilon$.
- A full-rank $n \times n$ submatrix H of F, determined by X and Y:

$$H = \begin{bmatrix} f(x_1y_1) & f(x_1y_2) & \cdots & f(x_1y_n) \\ f(x_2y_1) & f(x_2y_2) & \cdots & f(x_2y_n) \\ \vdots & \vdots & \ddots & \vdots \\ f(x_ny_1) & f(x_ny_2) & \cdots & f(x_ny_n) \end{bmatrix}$$

The entries of the matrix H are determined by making membership queries.

These data determine a *Hypothesis automaton* $\mathcal{A}$ as follows. Intuitively the states of $\mathcal{A}$ correspond to the rows of H, with the i-th row being the state reached after executing word x_i from the initial state. The columns can be considered as tests that distinguish different states.

Formally $\mathcal{A}$ has dimension n, initial-state vector $\alpha^\top := e_1^\top H$, the first row of H, and final-state vector $\eta := e_1$. Since H has full rank, for each $\sigma \in \Sigma$ we can define the transition matrix $M(\sigma)$ by the equation

$$HM(\sigma) = \begin{bmatrix} f(x_1\sigma y_1) & f(x_1\sigma y_2) & \cdots & f(x_1\sigma y_n) \\ f(x_2\sigma y_1) & f(x_2\sigma y_2) & \cdots & f(x_2\sigma y_n) \\ \vdots & \vdots & \ddots & \vdots \\ f(x_n\sigma y_1) & f(x_n\sigma y_2) & \cdots & f(x_n\sigma y_n) \end{bmatrix}$$

In each step of the algorithm an equivalence query is performed to determine whether $\mathcal{A}$ computes f. If not, a counterexample $w \in \Sigma^*$ is returned.

Proposition 22. *A counterexample z has a prefix $w\sigma$, where $\sigma \in \Sigma$ and $w \in \Sigma^*$, such that for some $i \in \{1, \ldots, n\}$ the assignment $X \leftarrow X \cup \{w\}$, $Y \leftarrow Y \cup \{\sigma y_i\}$ increases the rank of H by one.*

Proof. Say that automaton $\mathcal{A}$ is *correct* on a word $w \in \Sigma^*$ if

$$\alpha^\top M(w) = (f(wy_1), \ldots, f(wy_n)) \,. \tag{3}$$

Note that in this case $\mathcal{A}(w) = \alpha^\top M(w)\eta = f(w)$. It follows that $\mathcal{A}$ is not correct on z. Since it is clearly correct on the empty word, there must exist a prefix $w\sigma$ of z such that $\mathcal{A}$ is correct on w, but not on $w\sigma$. For such a w we have that (3) holds, but also

$$\alpha^\top M(w\sigma) \neq (f(w\sigma y_1), \ldots f(w\sigma y_n)) \,.$$

In particular, we can pick $i \in \{1, \ldots, n\}$ such that

$$\alpha^\top M(w\sigma)e_i \neq f(w\sigma y_i) \,. \tag{4}$$

Now consider the matrix H' defined by

$$H' = \begin{bmatrix} f(x_1y_1) & f(x_1y_2) & \cdots & f(x_1y_n) & f(x_1\sigma y_i) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ f(x_ny_1) & f(x_ny_2) & \cdots & f(x_ny_n) & f(x_n\sigma y_i) \\ f(wy_1) & f(wy_2) & \ldots & f(wy_n) & f(w\sigma y_i) \end{bmatrix}$$

$$\overset{(3)}{=} \begin{bmatrix} H & HM(\sigma)e_i \\ \alpha^\top M(w) & f(w\sigma y_i) \end{bmatrix} .$$

It remains to show that H' has rank $n+1$. By assumption H has rank n, so it suffices to show that the $(n+1)$-st row of H' cannot be expressed as a linear combination of the first n rows. Indeed, suppose for a contradiction that $u \in \mathbb{K}^n$ is such that $u^\top H = \alpha^\top M(w)$ and $u^\top HM(\sigma)e_i = f(w\sigma y_i)$. Then

$$f(w\sigma y_i) = u^\top HM(\sigma)e_i = \alpha M(w)M(\sigma)e_i \,,$$

which contradicts (4). □ □

The word w and suffix σy_i in the above proposition can be found using membership queries.

A Hoeffding's Inequality

Theorem 23 (Hoeffding's Inequality)**.** *Let $X_1, \ldots, X_m$ be independent random variables taking values in the interval $[a, b]$. Write $X = \frac{1}{m}\sum_{i=1}^m X_i$ and $\mu = E[X]$. Then for all $\varepsilon > 0$ we have*

$$\Pr(|X - \mu| > \varepsilon) \leq 2\exp\left(\frac{-2m\varepsilon^2}{(b-a)^2}\right) .$$

References

[1] A. Beimel, F. Bergadano, N. H. Bshouty, E. Kushilevitz, and S. Varricchio. Learning functions represented as multiplicity automata. *J. ACM*, 47:2000 (2000).

[2] A. Blumer, A. Ehrenfeucht, D. Haussler, and M.K. Warmuth. Learnability and the Vapnik-Cheronenkis dimension. *J. ACM*, 36:929–965 (1989).

[3] A. Chernikov and P. Simon. Externally definable sets and dependent pairs. *Israel Journal of Mathematics* , 194(1):409425 (2013).

[4] P. Goldberg and M. Jerrum. Bounding the Vapnik-Chervonenkis dimension of concept classes parameterized by real numbers. *Machine Learning* 18(2-3): 131-148 (1995).

[5] M. Grohe and Gy. Turán. Learnability and definability in trees and similar structures. *Theory Comput.* Syst. 37(1): 193-220 (2004).

[6] Y. Gurevich and S. Shelah. Spectra of Monadic Second-Order Formulas with One Unary Function. *Proceedings of 18th IEEE Symposium LICS*, IEEE Press, pages 291-300, (2003).

[7] H. R. Johnson and M. C. Laskowski. Compression schemes, stable definable families, and o-minimal structures. *Discrete Comput. Geom.*, 43(4):914926 (2010).

[8] R. Livni and P. Simon. Honest compressions and their application to compression schemes. In *Proceedings of COLT*, pages 77–92 (2013).

[9] M.C. Laskowski. Vapnik-Chervonenkis classes of definable sets. *J. London Math. Soc.* 45, 377-384 (1992).

[10] N. Littlestone and M. Warmuth. Relating data compression and learnability. Unpublished notes (1986).

[11] A. Pillay and C. Steinhorn. Definable sets in ordered structures. *Trans. Amer. Math. Soc.*, 295:565–592 (1986).

[12] W. Maass and Gy. Turán. On learnability and predicate logic. In *Proceedings of teh Bari-Ilan Symposium on the Foundations of Artificial Intelligence*, pages 75–85 (1995).

[13] M. Mohri, A. Rostamizadeh, and A. Talwalkar. *Foundations of Machine Learning*. MIT Press (2012).

[14] S. Moran and A. Yehudayoff. Sample compression schemes for VC classes. *J. ACM*, 63(3), 21:1-21:10 (2016)

[15] N. Sauer. On the densitiy of families of sets. *Journal of Combinatorial Theory, Series A*, 13:145–147 (1972).

[16] S. Shelah. Stability, the f.c.p., and superstability. *Ann. of Math. Logic 3*, 271-362 (1971).

[17] S. Shelah. A combinatorial problem: stability and order for models and theories in infinitary languages. *Pacific Journal of Mathematics*, 41:241–261 (1972).

[18] V.N. Vapnik and A.Y. Chervonenkis. On the uniform covergence of relative frequencies of events to their probabilities. *Theory of Prob. and is Appl.* 16, 264–280 (1971).

[19] H.E. Warren. Lower bounds for approximation by non-linear manifolds. *Trans. of the AMS*, 138:167–178 (1968).