Deciding the Vapnik-Červonenkis dimension is Σ_3^p-complete

Marcus Schäfer*
Department of Computer Science
University of Chicago
1100 East 58th Street
Chicago, Illinois 60637, USA
schaefer@cs.uchicago.edu

Abstract

Linial et al. *raised the question of how difficult the computation of the Vapnik-Červonenkis dimension of a concept class over a finite universe is. Papadimitriou and Yannakakis obtained a first answer using matrix representations of concept classes. However, this approach does not capture classes having exponential size, like monomials, which are encountered in learning theory. We choose a more natural representation, which leads us to redefine the* VC DIMENSION *problem. We establish that* VC DIMENSION *is* Σ_3^p*-complete, thereby giving a rare natural example of a* Σ_3^p*-complete problem.*

1 Introduction

An important contribution of mathematics to computer science has been the notion of the Vapnik-Červonenkis dimension, or VC dimension for short. The notion originated in probability theory [14], where Vapnik and Červonenkis used it to prove uniform convergence of empirical distributions (see [5]). Their idea was successfully taken up by Blumer, Ehrenfeucht, Haussler and Warmuth and applied to learning theory [4]. Later applications include finite automata [6], computability theory [3, 7] and computational geometry [8], where an effective version of the Vapnik-Červonenkis dimension was suggested.

Over a given universe U a *concept* is a subset of the universe, and a *concept class* is a collection of concepts. Then the *Vapnik-Červonenkis dimension* of a concept class is defined to be the cardinality of the largest set $F \subset U$ such that any subset of F can be obtained by intersecting F with a concept in $\mathcal{C}$.

Let us have a closer look at the VC dimension in learning theory. Blumer *et al.* showed in [4] that the VC dimension is a direct measure of the difficulty of learning a concept class. This gave rise to interest in the actual computation of the VC dimension, and led to the question, what is the complexity of computing the VC dimension of a concept class? This question was first asked by Linial, Mansour and Rivest in [9]. To make it meaningful, we restrict ourselves to finite universes U. This is sometimes called the discrete VC dimension problem.

Since the universe is finite, so is the concept class, say $\mathcal{C} = \{C_1, \ldots, C_m\}$. Typically the concepts in $\mathcal{C}$ can be generated in some effective manner. To illustrate this, consider M_n, the monomials in n variables, a standard example taken from learning theory (see [1]). If we identify a monomial with the assignments in $U = \{0,1\}^n$, which make it true, then every monomial is a concept, and M_n is a concept class as defined above. How do we represent this class? The concept class M_n can be generated by a polynomial-size circuit $C(i,x)$, which takes as first input an encoding $i \in \{0,1\}^n$ of a monomial in M_n, and accepts the second input $x \in \{0,1\}^n$, if the monomial encoded by i is true on assignment x. Then $M_n = \{\{x : C(i,x) \text{ accepts}\} : i \in \{0,1\}^n\}$. We can now ask, how difficult it is to compute the VC dimension of M_n given the circuit C as its representation.

Looking at other examples, like k-DNFs in less than n variables, circuits with less than n gates, accepting paths of (precise) **NP**-machines on inputs of length n, etc, shows that each of these classes can be generated by a single circuit; that is the concept classes we are normally interested in, have an effective structure, which allows for a representation that has polynomial size in the original parameters (n in the examples above). Furthermore *any* concept class over a finite universe can be represented by a circuit, if needs be by hardwiring the matrix representation of the concept class into the circuit, although this will lead to a blow-up of the circuit.

This corroborates the view that concept classes over

*Partially supported by NSF Grant CCR 92-53582

finite universes are naturally represented by generating circuits, and we will therefore define VC DIMENSION to be the problem of deciding whether a concept class, represented by a circuit, has VC dimension at least k. A direct argument shows that VC DIMENSION is in $\mathbf{\Sigma_3^P}$, and as the main result we prove that it is complete for this class. So in addition to classifying the VC DIMENSION problem, we present a rare natural example of a $\mathbf{\Sigma_3^P}$-complete problem.

When Linial *et al.* first asked for a classification of the VC dimension problem, they restricted representations of concept classes to incidence matrices. This is oblivious to the fact that natural concept classes typically have an effective parameterization, which allows for representations that are exponentially smaller than the matrix representation, as exemplified above in the case of monomials. I.e. the restriction to matrix representations leads to an exponential blow-up of the input. Let us call Linial *et al.*'s approach the MATRIX VC DIMENSION problem. They observed that MATRIX VC DIMENSION can be solved in time $O(n^{\log n})$, where n is the size of the matrix. Later Papadimitriou and Yannakakis in [11] defined a new complexity class **LOGNP** and showed the problem to be complete for this class. Similarly Shinohara [13] classified MATRIX VC DIMENSION with regard to two other classes.

2 Computing the VC dimension

Most of the notation is standard and can be found in [10]. We will let Σ denote the alphabet $\{0,1\}$. The notation $x[j..k]$ is the substring of x that starts at the jth bit and ends with the kth bit. So x is the same as $x[1..n]$, if n is the length of x. Similarly the jth bit of x will be $x[j]$.

Definition 2.1 *Let $\mathcal{C}$ be a concept class over a universe U. Then the Vapnik-Červonenkis dimension (VC dimension) of $\mathcal{C}$, written as $VC(\mathcal{C})$, is the largest number n for which there is a set $F \subseteq U$ of cardinality n, such that any subset S of F can be written as the intersection of F with a concept C in $\mathcal{C}$, i.e. $S = C \cap F$. If no such number exists, the Vapnik-Červonenkis dimension is said to be infinite.*

As explained in the introduction, we will ask for the VC dimension of a concept class generated by a circuit. Let us state the problem VC DIMENSION formally.

VC DIMENSION
Instance: A circuit C with $l+m$ input gates, and a positive integer k.
Question: Define $C_i := \{x \in \Sigma^m : C(i,x) = 1\}$ for $i \in \Sigma^l$, and the the concept class $\mathcal{C} := \{C_i : i \in \Sigma^l\}$ over Σ^m. Is $VC(\mathcal{C}) \geq k$?

From now on we will write $VC(C)$ for $VC(\mathcal{C})$, if $\mathcal{C}$ is the concept class generated by circuit C in the manner described above.

The definition of the VC dimension immediately implies that the concept classes we are interested in, namely finite concept classes, have finite VC dimension. More precisely the following fact holds.

Fact 2.2 *A concept class comprising at most 2^m concepts, has VC dimension at most m.*

A glance at the definition of the VC dimension will convince the reader that VC DIMENSION is easily solvable in **PSPACE**. In fact we can do better than that.

Lemma 2.3 VC DIMENSION *is in $\mathbf{\Sigma_3^P}$.*

Proof Suppose we are given a circuit C and an integer k (in binary). The circuit produces at most $2^{|C|}$ sets C_i, where $|C|$ is the length of the encoding of the circuit C. By the preceding fact, the VC dimension of C is bounded by $|C|$. So if $k \geq |C|$, we reject. We may therefore assume that k is less than $|C|$. Let us restate more formally what it means for a circuit to have VC dimension at least k.

$$\begin{aligned} VC(C) \geq k \quad \iff \quad & (\exists x_1,\ldots,x_k \in \Sigma^m) \\ & (\forall s \in \Sigma^k)(\exists i \in \Sigma^l) \\ & (\forall j \in \{1,\ldots,k\})[C(i,x_j) = s[j]] \end{aligned}$$

The innermost quantifier is bounded by k and thus polynomial-time computable in $|C|$. Since m, k and l are bounded by $|C|$, the formula can be decided in $\mathbf{\Sigma_3^P}$. Therefore VC DIMENSION is in $\mathbf{\Sigma_3^P}$. □

More importantly we can prove that this classification is optimal.

Theorem 2.4 VC DIMENSION *is $\mathbf{\Sigma_3^P}$-complete.*

Proof We will show how to reduce QSAT$_3$, the standard $\mathbf{\Sigma_3^P}$-complete problem to VC DIMENSION. Together with the preceding lemma, this proves the theorem.

Suppose we are given an instance of QSAT$_3$, that is a formula Ψ of the form $(\exists a)(\forall b)(\exists c)\Phi(a,b,c)$, where Φ is a boolean expression in CNF and the variables in Φ are partitioned into three sets. The strings a, b and c are assignments to variables in these sets. By adding new variables, we can achieve that all sets contain the same number of variables, say $a, b, c \in \Sigma^k$.

We will now construct a circuit C that establishes the reduction of QSAT$_3$ to VC DIMENSION.

Set $l := \lceil \log(k+1) \rceil$. Consider inputs $i = uvv'w \in \Sigma^{k+k+1+k}$ and $x = yz \in \Sigma^{k+l}$. Note that we will use z as an integer rather than as a string. Define C on inputs i and x as follows:

$$C(i,x) := \begin{cases} i[k+z] & \text{if } 1 \le z \le k+1, u = y, \text{and} \\ & \Phi(uvw) \text{ is true} \\ 0 & \text{else} \end{cases}$$

Obviously C can be realized by a circuit having polynomial size in the length of Ψ. We will now prove that

$$(\exists a)(\forall b)(\exists c)\Phi(a,b,c) \iff VC(C) \ge k+1,$$

which shows that we have in fact reduced QSAT$_3$ to VC DIMENSION.

($\Rightarrow$) Suppose $(\exists a)(\forall b)(\exists c)\Phi(a,b,c)$ is true. Fix $a \in \Sigma^k$ such that $(\forall b)(\exists c)\Phi(a,b,c)$ is true. For all $j \in \{1,\ldots,k+1\}$ set $x_j := a\underline{j}$, where $\underline{j}$ is an l-bit binary representation of j. Let s be any string in Σ^{k+1}. By assumption for $b := s[1..k]$ there exists $c \in \Sigma^k$, for which $\Phi(a,b,c)$ is true. Let i be the string asc, then $C(i,x_j)$ is just $s[j]$, and therefore $VC(C) \ge k+1$.

($\Leftarrow$) Assume now that $VC(C) \ge k+1$, i.e. there are $x_1,\ldots,x_{k+1} \in \Sigma^{k+l}$ such that $(\forall s \in \Sigma^{k+1})(\exists i \in \Sigma^{k+(k+1)+k})(\forall j \in \{1,\ldots,k+1\})[C(i,x_j) = s[j]]$. This implies that the x_j have to be pairwise different. By ordering them lexicographically, we may assume that $x_1 < x_2 < \ldots < x_{k+1}$. Then the following holds:

- All x_j agree on the first k bits.
- The number encoded by the last l bits of x_j is j (for all j).

To prove the first claim, suppose for contradiction that x_j and x_h do not agree on their first k bits. Choose a vector s such that $s[j] = 1$ and $s[h] = 1$. By assumption there is an i such that $C(i,x_j) = 1$ and $C(i,x_h) = 1$. But in this case i would have to agree with x_j and x_h on the first k bits, contradicting the assumption on x_j and x_h. A similar argument shows that the number encoded by the last l bits of x_j is contained in the set $\{1,\ldots,k+1\}$. Suppose there were two strings x_j and x_h with $j \ne h$, which encode the same number. Since x_j and x_h agree on their first k bits, this means $x_j = x_h$, which contradicts the fact that the strings are pairwise different. So the $k+1$ vectors x_j encode $k+1$ different integers in the set $\{1,\ldots,k+1\}$ in ascending order. Thus x_j has to encode j.

With this information on the x_j we can complete the proof of the theorem. Since all x_j agree on the first k bits, we can define a as the first n bits of any x_j. Let $b \in \Sigma^k$ be given. Set $s := b1$. By assumption there is an $i = uvv'w \in \Sigma^{k+k+1+k}$ such that for all j: $C(i,x_j) = s[j]$. Since $s[k+1]$ is 1, we know that the first case in the definition of $C(i,x_{k+1})$ must apply, and therefore $\Phi(uvw)$ is true, and $u = a$. Furthermore $C(i,x_j) = s[j]$ for all j, and thus $v = b$. Setting $c := w$ yields that $\Phi(a,b,c)$ is true and hence $(\exists a)(\forall b)(\exists c)\Phi(a,b,c)$ holds. □

Remark To round up the picture let us consider two variants of the VC dimension problem in computability theory. For this we remove the restriction that the universe be finite, and assume it to be ω, the set of natural numbers. We shall call a concept class $\mathcal{C}$ (uniformly) computable, if $\mathcal{C} = \{C_1,\ldots\}$ and membership in C_i can be decided uniformly in i. Similarly we call $\mathcal{C}$ (uniformly) computably enumerable (or c.e. for short), if the C_i can be enumerated uniformly in i.

Computable concept classes capture geometric concept classes very well, whereas c.e. concept classes occur naturally in the context of automata and languages.

- Deciding whether a computable concept class has finite VC dimension is $\mathbf{\Sigma_2}$-complete,
- Deciding whether an c.e. concept class has finite VC dimension is $\mathbf{\Sigma_3}$-complete,

Proofs of both results are not very hard. The second result was proved independently in [15].

3 Further Research

We have proved that deciding the VC dimension of a concept class described by a circuit is $\mathbf{\Sigma_3^P}$-complete. But what is the complexity of the associated function problem, i.e. how difficult is it to compute the function $VC : C \mapsto VC(C)$? Using VC as an oracle, $\mathbf{\Sigma_3^P}$ can be decided in polynomial time. A simple construction shows that this remains true, if instead of VC we only get an approximation of VC to within an additive constant.

What happens if instead of additive constants, we consider multiplicative approximations?

Definition 3.1 *We say the VC dimension is $h(n)$-approximated by a function f if*

$$1/h(|C|) < f(C)/(VC(C)+1) < h(|C|).$$

The following is an easy result.

Theorem 3.2 *If the VC dimension is $n^{1/2-\varepsilon}$-approximated by a function f for some $\varepsilon > 0$, then* $\mathbf{NP} \subseteq \mathbf{P}^f[1]$, *i.e.* **NP** *can be decided in polynomial time with one oracle query to f.*

Proof (Sketch) Let Φ be a formula of length n. Define $l := \lceil \log n \rceil$, and $k := 2^l$. Construct a circuit C with inputs $i = ay \in \Sigma^{n+k}$ and $x \in \Sigma^l$ that works as follows:

$$C(i,x) := \begin{cases} y[x] & \text{if } \Phi(a) \text{ is true} \\ 0 & \text{else} \end{cases}$$

If Φ is satisfiable, then $VC(C) = k \geq n$, else $VC(C) = 0$. □

So in particular we can't approximate VC up to a constant factor in polynomial time, unless $\mathbf{P} = \mathbf{NP}$. It would be very interesting to strengthen the case of constant factors to $\mathbf{\Sigma}_3^{\mathrm{P}}$, and get a strong non-approximability result. Alternately (but less likely), determining the VC dimension up to a multiplicative factor might be easier than computing the exact value. Settling this question either way might lead to some new insights into the combinatorial behavior of VC.

Acknowledgments I'd like to thank Lance Fortnow for suggesting the problem to me, and making a lot of helpful comments, as did Ashish Naik, Dieter van Melkebeek and Martin Kummer.

References

[1] Martin Anthony, Norman Biggs. *Computational Learning Theory.* Cambridge University Press, 1992.

[2] Patrick Assouad. Densité et dimension. *Ann. Inst. Fourier, Grenoble*, 33,3:233–282, 1983.

[3] Richard Beigel, Martin Kummer, Frank Stephan. Quantifying the amount of verboseness. *Information and Computation*, 118:73–90,1995.

[4] Anselm Blumer, Andzej Ehrenfeucht, David Haussler, Manfred K. Warmuth. Learnability and the Vapnik-Chervonenkis dimension. *Journal of the Association for Computing Machinery*, 36,4: 929–965, 1989.

[5] Richard Mansfield Dudley. A course on empirical processes. In: *École d'Été de Probabilités de Saint-Flour XI I - 1982.* Lecture Notes in Mathematics 1097, Springer-Verlag, Berlin, 2–142, 1984.

[6] Yoshiyasu Ishigami, Sei'ichi Tani. The VC dimensions of Finite Automata with n States. *Proceedings of the 4th Workshop in Algorithmic Learning Theory*, 328–341, 1993.

[7] Susanne Kaufmann, Martin Kummer. On a Quantitative Notion of Uniformity. to appear in: *Fundamenta Informaticae*, an extended abstract appeared in *Proceedings MFCS'95, LNCS 969*, 169–178, Springer-Verlag, 1995.

[8] Martin Kummer, Marcus Schäfer. Computability of Convex Sets. *Proceedings of the 12th Symposium on Theoretical Aspects of Computer Science*, 550–561, 1995.

[9] Nathan Linial, Yishay Mansour, Ronald L. Rivest. Results on learnability and Vapnik-Červonenkis dimension. *Proceedings of FOCS*, 120–129, 1988.

[10] Christos H. Papadimitriou. *Computational Complexity.* Addison-Wesley, 1994.

[11] Christos H. Papadimitriou, Mihalis Yannakakis. On Limited Nondeterminism and the Complexity of the V-C Dimension. *Proceedings of the 8th Conference on Structure in Complexity Theory*, 12–18, 1993.

[12] Akito Sakurai. On the VC dimension of Depth Four Threshold Circuits and the Complexity of Boolean-valued Functions. *Proceedings of the 4th Workshop in Algorithmic Learning Theory*, 279–287, 1993.

[13] Ayumi Shinohara. Complexity of Computing Vapnik-Červonenkis Dimension. *Proceedings of the 4th Workshop in Algorithmic Learning Theory*, 279–287, 1993.

[14] V. N. Vapnik, A. Ya. Červonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory Probab. Appl.*, 16:264–280, 1971.

[15] Stephan Wehner. *Zur Komplexität des Numerierens.* Thesis at the Fakultät für Informatik, Universität Karlsruhe, November 1990.