INFORMATION AND COMPUTATION **115**, 248–292 (1994)

Predicting $\{0, 1\}$-Functions on Randomly Drawn Points*

D. HAUSSLER

Department of Computer and Information Sciences,
University of California, Santa Cruz, California 95064

N. LITTLESTONE

NEC Research Institute, 4 Independence Way, Princeton,
New Jersey 08540

AND

M. K. WARMUTH

Department of Computer and Information Sciences,
University of California, Santa Cruz, California 95064

We consider the problem of predicting $\{0, 1\}$-valued functions on $\mathbf{R}^n$ and smaller domains, based on their values on randomly drawn points. Our model is related to Valiant's PAC learning model, but does not require the hypotheses used for prediction to be represented in any specified form. In our main result we show how to construct prediction strategies that are optimal to within a constant factor for any reasonable class F of target functions. This result is based on new combinatorial results about classes of functions of finite VC dimension. We also discuss more computationally efficient algorithms for predicting indicator functions of axis-parallel rectangles, more general intersection closed concept classes, and halfspaces in $\mathbf{R}^n$. These are also optimal to within a constant factor. Finally, we compare the general performance of prediction strategies derived by our method to that of those derived from methods in PAC learning theory. © 1994 Academic Press, Inc.

1. INTRODUCTION

Let F be a class of $\{0, 1\}$-valued functions on a fixed domain X. The domain can be finite, countably infinite, or $\mathbf{R}^n$ for some $n \geqslant 1$. We consider the problem of predicting the value of an unknown target function $f \in F$ on

* We gratefully acknowledge the support of ONR Grants N00014-86-K-0454 and N00014-85-K-0445, and DARPA Grant AFOSR-89-0506. Part of this research was done while the second author was in Santa Cruz, and part while he was at Harvard.

0890-5401/94 $6.00

a randomly drawn point $x_t \in X$, given the value of f on randomly drawn points $x_1, ..., x_{t-1} \in X$. A rule for making such predictions is called a *prediction strategy for F.* A prediction strategy can be either deterministic or randomized; i.e., the prediction on x_t may be determined by the value of f on the points $x_1, ..., x_{t-1}$, or it may also depend on the value of some other independent random event.

Given feedback from some external agent that indicates whether or not the prediction is correct, a prediction strategy can be iterated indefinitely. In each iteration, a random point is drawn, a prediction is made based on feedback from previous iterations, and new feedback is received. Each such iteration is called a *trial.*

Let us assume that an adversary who knows our prediction strategy is allowed to choose the target function $f \in F$ and the probability distribution P on X. All points are then drawn independently according to P. We are interested in finding a prediction strategy for F that, for each $t \geqslant 1$, minimizes the probability that an incorrect prediction (*mistake*) is made on trial t. Since an adversary chooses the target function and the distribution, our objective is to minimize this probability in the worst case, over all $f \in F$ and distributions P on X. We use $\hat{\mathbf{M}}(t)$ to denote this worst case probability.

This prediction model is based on the non-probabilistic prediction model studied in [L87]. There the points $x_1, x_2, ...$ are directly selected by the adversary and the object is to make the smallest total number of mistakes. A closely related model is defined in [A87], and is known as learning from equivalence queries with counterexamples. Our probabilistic variant of this model has the advantage that it is applicable even in situations where this worst case total number of mistakes is infinite, as often occurs when the domain X is infinite (e.g., $\mathbf{R}^n$, or Σ^*, for some finite alphabet Σ).

Our probabilistic prediction model is motivated by, and closely related to, the learnability model of Valiant [Val84, BEHW87, 89, AL87, R87, KLPV87], which is often called the *Probably Approximately Correct* (PAC) learning model [A87]. In the PAC model the learning algorithm must output a representation of a hypothesis in some hypothesis class H when given random examples of some unknown target function in F. The hypothesis must (with high probability) be a good approximation to the target function in the sense that (with high probability) it correctly predicts the value of the function on further examples drawn from the same distribution. As in our prediction model, these probabilities are calculated with respect to the worst case over all target functions in F and all distributions on the domain.

Prediction strategies and learning algorithms both form hypotheses, although the hypothesis of a prediction strategy is only defined implicitly. Given the value of the unknown function f on points $x_1, ..., x_{t-1}$ from the previous $t-1$ trials, the hypothesis h formed by a deterministic prediction strategy is defined by how the strategy would predict on each possible x that might be presented to it in the tth trial: $h(x) = 1$ if the prediction is 1, $h(x) = 0$ otherwise. Randomized prediction strategies form "randomized hypotheses" (see, e.g., [Sch90]). Hence our prediction model is essentially a streamlined version of the PAC model, in which the requirement that hypotheses be in a certain hypothesis class and be represented in a specified form has been dropped, and only predictive performance is measured. We also simplify the model by considering only one probability (i.e., the probability of a mistake) instead of two (i.e., the probability that the algorithm produces a hypothesis with small probability of a mistake). The exact relationship between our prediction model and the PAC model is discussed further in Sections 4 and 5 below, and in detail in [HKLW91, PW90].

Summary of Results

As in [BEHW89, L87], our results are of two types: the first type consists of results that indicate how well a learner can possibly do from the available information, if computational complexity is not an issue; the second type considers what can be done in polynomial time.

In our main result (Theorem 2.3) we show that, ignoring computational complexity, for any reasonable[1] class F of target functions we can construct a prediction strategy for F for which the worst case probability of a mistake on trial t, $\hat{\mathbf{M}}(t)$, is within a constant factor of the best possible; we call such a prediction strategy *essentially optimal*. Specifically, when this prediction strategy is applied to target functions from F, then $\hat{\mathbf{M}}(t)$ is at most $\mathrm{VCdim}(F)/t$, for any $t \geqslant 1$. Here $\mathrm{VCdim}(F)$ denotes the *Vapnik–Chervonenkis dimension of* F as defined in [HW87] (following [VC71, Vap82]). From a result in [EHKV89], it can be shown that if F is nontrivial (see Section 3) and $\mathrm{VCdim}(F)$ is finite, then for any prediction strategy for F, $\hat{\mathbf{M}}(t)$ is $\Omega(\mathrm{VCdim}(F)/t)$, and if $\mathrm{VCdim}(F)$ is infinite, results from [BEHW89] imply that $\hat{\mathbf{M}}(t) \geqslant \frac{1}{2}$, for all $t \geqslant 1$ (Theorem 3.1). This shows that the upper bound is tight to within a constant factor.

Note that this essentially optimal prediction strategy has a pleasant property when applied to a class F of finite VC dimension. For any target concept in F and any distribution on X, the expected number of mistakes during the second half of any sequence of trials is at most $\sum_{i=\lceil t/2 \rceil + 1}^{t} \mathrm{VCdim}(F)/i < \ln(2)\,\mathrm{VCdim}(F)$. For example, if $\mathrm{VCdim}(F) = 4$ then when you iterate the strategy for 100 trials, the expected number of mistakes during the last 50 trials is at most 3, and when you iterate it for 1,000,000 trials the expected number of mistakes during the last 500,000 trials is still at most 3. Further results on cumulative mistake bounds are given in Sections 2 and 3.

[1] When $X = \mathbf{R}^n$ we impose certain measurability constraints on F as in [BEHW89].

If one can tell in polynomial time whether or not there is any function in F that is consistent with a sequence of examples, then our general essentially optimal prediction strategy runs in time polynomial in the trial index t, where the exponent is proportional to VCdim(F). This prediction strategy is not practical when VCdim(F) is large. However, we can improve this time bound considerably for many important classes of functions, including indicator functions for halfspaces and axis-parallel rectangles in $\mathbf{R}^n$. We describe essentially optimal prediction strategies for these classes that run in time polynomial in both t and VCdim(F) (see Examples 2.1 through 2.4).

In Section 4 we compare our method of constructing prediction strategies with another technique for constructing prediction strategies based on the learning results of [BEHW89]. The latter technique constructs a hypothesis that is consistent with the previous trials and chosen from a fixed class H of possible hypotheses, and uses this hypothesis to make its prediction. We show that for any H, the probability of making a mistake at trial t for such a prediction strategy is $O(\log \alpha/\alpha)$, where $\alpha = t/\text{VCdim}(H)$ (Theorem 4.1). Here we use the techniques of Vapnik and Chervonenkis [Vap82]. Note that this bound depends on the VC dimension of the hypothesis space, whereas the optimal bound depends on the VC dimension of the target class. Any consistent algorithm must have $\text{VCdim}(H) \geqslant \text{VCdim}(F)$.

This result gives a useful upper bound, since many prediction strategies that are derived directly from learning algorithms are of the above type. However, even if $\text{VCdim}(H) = \text{VCdim}(F)$, for sufficiently large t this bound is still worse than that of the essentially optimal strategy given in Section 2 by a factor proportional to $\log t/\text{VCdim}(F)$. We show that this performance gap is real by exhibiting, for every $d \geqslant 1$, a class of functions F with $\text{VCdim}(F) = d$, a target function $f \in F$, a consistent prediction strategy that always chooses a hypothesis from F, and for each t a distribution on the domain of f, such that the probability of a mistake on trial t is $\Omega(\log \alpha/\alpha)$, where $\alpha = t/d$ (Theorem 4.2).

As mentioned above, it is easy to see that any PAC-style learning algorithm defines a prediction strategy, and any prediction strategy defines a PAC-style learning algorithm. In the latter case, after processing the given batch of examples the algorithm outputs the current state of the prediction strategy as a representation of the hypothesis, so the hypothesis class used by this algorithm is in general different from the class of target functions. In Section 5 we exhibit a PAC-style learning algorithm that, for any target function in F and any distribution on the domain, produces, with probability at least $1-\delta$, a hypothesis with error at most ε using $O((\text{VCdim}(F)/\varepsilon) \log(1/\delta))$ independent random training examples. (As above, the "error" of a hypothesis is the probability that it disagrees with the target function on a randomly drawn point.) This algorithm first produces a small set of hypotheses by applying the optimal prediction strategy discussed in Section 2 to independent batches of examples. Then it uses additional examples to select a good hypothesis from this set. The sample size bound for this algorithm is, for some choices of ε and δ, better than the $O((\mathrm{VCdim}(F)/\varepsilon)\log(1/\varepsilon)+(1/\varepsilon)\log(1/\delta))$ bound given in [BEHW89] for learning algorithms that produce consistent hypotheses in F, which was the previous best general bound. It is still an open problem to find a general learning algorithm that meets the $\Omega(\mathrm{VCdim}(F)/\varepsilon)+(1/\varepsilon)\log(1/\delta)$ lower bound established in [EHKV89].

In Section 6 we discuss new research directions and give a number of open problems.

Overview of Methods Used in Main Result

In the model that we consider, the input to a prediction strategy is in the form of a sequence of independent random points selected according to an arbitrary, unknown distribution. Lack of knowledge about the distribution can make the performance of the strategy difficult to analyze. However, consider a fixed sequence of points $\bar{x}=(x_1, ..., x_t)\in X^t$. Whatever the underlying distribution, the process of independent random selection induces a uniform distribution on the permutations of this sequence; one is as likely to draw one permutation of $\bar{x}$ as another. As in [VC71, Vap82], we use this observation to obtain performance bounds for arbitrary distributions from combinatorial arguments about permutations of sequences. Seidel also uses similar techniques to analyze the expected performance of algorithms for problems in computational geometry and other areas of computer science [Sei91].

For a deterministic prediction strategy, let the *permutation mistake bound* $\hat{\hat{\mathbf{M}}}(t)$ denote the supremum, over all $f\in F$ and all sequences $\bar{x}$ of t points in X, of the fraction of permutations of $\bar{x}$ for which the prediction strategy makes a mistake predicting the value of f on the last point, given its value on the previous points. The bound is also defined for randomized strategies using a natural generalization of this definition (see Section 2). For many prediction strategies, the permutation mistake bound can be calculated by a simple counting argument. Using the observation above, it is easy to show that $\hat{\mathbf{M}}(t)\leqslant\hat{\hat{\mathbf{M}}}(t)$, so this function provides a convenient upper bound on our primary performance measure. All our bounds for $\hat{\mathbf{M}}(t)$, including those for the computationally efficient prediction strategies that we give for halfspaces and axis-parallel rectangles and general intersection closed concept classes, are obtained by bounding $\hat{\hat{\mathbf{M}}}(t)$.

We now describe the principles that the essentially optimal strategy is based on. Let F be a class of $\{0, 1\}$-valued functions on X. Two functions in F can be considered equivalent with respect to the fixed sequence $\bar{x}$ if their values agree on all the points in $\bar{x}$; hence $\bar{x}$ naturally partitions F into a set of at most 2^t equivalence classes. VCdim(F) can be defined as the largest t such that there exists a sequence $\bar{x} = (x_1, ..., x_t)$ that induces 2^t distinct, nonempty equivalence classes with respect to F.

Suppose that points $x_1, ..., x_{t-1}$ are labeled with the values of an unknown $f \in F$, and your task is to predict the value of f on x_t. If you are lucky then there is only one nonempty equivalence class consistent with the labels of $x_1, ..., x_{t-1}$ and thus the value of f on x_t is determined. However, if there remains a pair of consistent nonempty equivalence classes (one for $f(x_t) = 0$ and the other for $f(x_t) = 1$), then the situation is ambiguous. We try to resolve this ambiguity in a way that will minimize $\hat{\mathbf{M}}(t)$.

To do this we construct, for the given F and $\bar{x}$, a graph G called the *1-inclusion graph* [B72, AHW87] (see also [F89]), whose nodes are the equivalence classes of F induced by $\bar{x}$ and whose edges represent possible pairs of equivalence classes that may remain at trial t for some permutation of $\bar{x}$. By directing the edges of G, we can represent a deterministic prediction strategy for the permutations of $\bar{x}$. We show that if there is a way of directing the edges of every 1-inclusion graph derived from F so that the maximum outdegree of any node is k, then this defines a prediction strategy with $\hat{\mathbf{M}}(t) \leqslant k/t$.

For example, consider the class of indicator functions of closed intervals over $\mathbf{R}$; each function in the class is 1 on some closed interval and 0 elsewhere. This function class has VC dimension 2. Now consider the equivalence classes induced by this function class on any sample of t distinct points $x_1 < x_2 < \cdots < x_t$. We illustrate the 1-inclusion graph for these equivalence classes in Fig. 1. Each equivalence class is represented by the subset of $\{x_1, ..., x_t\}$ on which the functions in the equivalence class are 1. Though the degree of the 1-inclusion graph grows in proportion to t (the degree of the top node is t), the outdegree of each node will be at most two if all edges in the illustrated graph are directed upward. It turns out that this defines the simple prediction strategy of guessing 0 on x unless x lies between two points known to have value 1. More general versions of this strategy are given in Examples 2.1, 2.2, and 2.3.

In Section 2 we define a probabilistic prediction strategy by labeling the ends of each edge of the 1-inclusion graph with a probability such that the two probabilities on each edge sum to one. These probabilities represent a randomized rule for resolving the ambiguous situation represented by that edge. Using the max-flow/min-cut theorem, we can show that by choosing the edge probabilities appropriately, this leads to a randomized prediction strategy with $\hat{\mathbf{M}}(t) \leqslant \text{maxdens}_t(F)/t$, where $\text{maxdens}_t(F)$ is the maximum density (number of edges divided by number of nodes) of any subgraph of a 1-inclusion graph of F for at most t instances. Our key combinatorial lemma (Lemma 2.4) shows that for any function class F the density of any

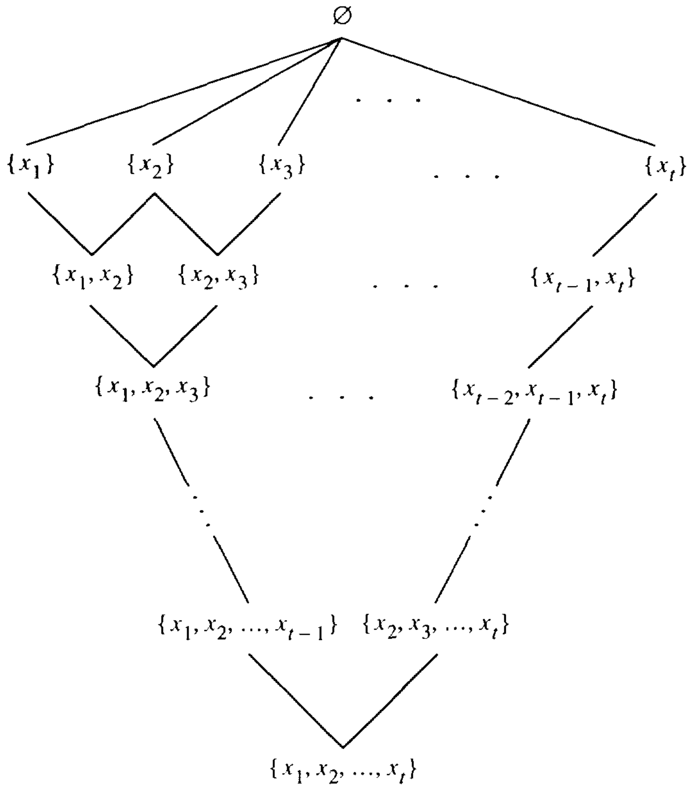


FIG. 1. 1-inclusion graph for a function class of VC dimension 2.

1-inclusion graph derived from F is less than VCdim(F), leading to a randomized prediction strategy with $\hat{\mathbf{M}}(t) \leqslant \hat{\hat{\mathbf{M}}}(t) < \mathrm{VCdim}(F)/t$. A slightly modified argument leads to a deterministic prediction strategy for which $\hat{\hat{\mathbf{M}}}(t) \leqslant \mathrm{VCdim}(F)/t$.

Relation to Other Work

This work can be viewed as an extension of the general line of research initiated by Valiant in [Val84], aimed at developing a useful theoretical foundation for the analysis of machine learning algorithms used in artificial intelligence, robotics, and other areas. The approach here differs from the approach emphasized in the Valiant PAC model in that we focus on the predictive performance of learning algorithms, whereas much of the work in the PAC model has focused on learning representations, that is, on finding hypotheses that approximate the target function and are represented in a particular way. The learning problem has often been taken to be one of finding a hypothesis from the target class itself. The difficulty with this approach is that for many seemingly simple target classes it is NP-hard to learn the target class using hypotheses from the same class [PV88].[2] Here we work with a prediction model in which it is possible to explore issues of computationally practical learnability without being encumbered by restrictions on the hypothesis space (other than computational restrictions requiring the hypotheses to be evaluatable in polynomial time). As a consequence, the predictability model forms a more appropriate basis for obtaining hardness results for learning problems. If it can be shown that there are no efficient and effective prediction strategies for a class of functions F, then this implies that F is not polynomially learnable by forming hypotheses from any polynomially evaluatable hypothesis class. Such hardness results have been obtained (modulo certain cryptographic assumptions) for several classes of functions in [KV92, PW90]. The latter paper introduces a general notion of prediction preserving reducibility among learning problems, akin to the usual notions of reducibility among combinatorial problems.

Since the appearance of [A87, L87] and the conference version of this paper, numerous authors, apart from those mentioned above, have made further contributions to the study of prediction strategies in the sense that we have defined them. These include the papers [A90, B90, L89a, b, LW94, MT92, MT94a, b] that focus on the non-probabilistic mistake bound or equivalence query model from [A87, L87], giving quantitative results on mistake bounds and comparisons between this and other learning models. The papers [F95, GKS95, Sch90] focus on what is known as "weak learning," in which a learning algorithm or prediction strategy is required only to do slightly better than random guessing. In [F89] the existence of space efficient prediction strategies is investigated (see also [H88b, Sch90]). These are prediction strategies that save only very little information from previous trials. The papers [GRS93, HSW90, 92] give algorithms with good expected mistake bounds for a variety of concept classes, including those derived from binary relations, linear orders, integer lattices, subspaces of a vector space, general intersection closed classes (see Example 2.2), and nested differences of such classes. Prediction strategies like those defined here have also recently been studied from a Bayesian or "average case" perspective in [HKS94, OH91] (in contrast to the minimax approach taken here), the former paper building directly on this work in some of its results. The extent of this work provides further evidence that

[2] For example, in [PV88] it is shown that the class of Boolean functions represented by 2-term DNF expressions is not polynomially learnable by any algorithm that must represent its hypothesis as a 2-term DNF (unless $\mathbf{NP} = \mathbf{RP}$); however, this class is polynomially learnable if we allow hypotheses represented as 2-CNF expressions.

the analysis of prediction strategies can make useful contributions to computational learning theory.

Finally, we have introduced a new combinatorial property of target classes of finite VC dimension in Lemma 2.4. This result has led to other combinatorial properties of VC classes [H95], which have applications in the theory of empirical processes [T92]. The methods used in obtaining this result are similar to those used in combinatorial geometry to bound the number of cells, faces, edges, etc., in an arrangement of hyperplanes (see, e.g., [E87]); hence, this further underscores the unexpectedly close relationship between geometry and learning (see also [HW87, W88, CF88, CW89, KPG92, MSW90] for applications of the VC dimension in geometry.) The general, essentially optimal prediction strategy we give is the first learning algorithm that directly exploits deeper combinatorial properties of target classes of finite VC dimension to make its predictions; others have simply made predictions by forming arbitrary consistent hypotheses from classes of small VC dimension. This further exploration of the structure of classes of finite VC dimension and its application to learning is one of the main contributions of this paper.

General Notation. We denote by X a set called the *domain*. We assume X is either finite, countably infinite, or $\mathbf{R}^n$ for some $n \geqslant 1$, where $\mathbf{R}$ denotes the real numbers. By F we denote a nonempty class of $\{0, 1\}$-valued functions on X. We call such a class of functions a *concept class.*[3]

For $t \geqslant 1$, $\bar{x} = (x_1, ..., x_t) \in X^t$, and $f \in F$, $\mathrm{sam}(\bar{x}, f) = ((x_1, f(x_1)), ..., (x_t, f(x_t)))$. We call $\mathrm{sam}(\bar{x}, f)$ the *sample of f generated by $\bar{x}$*, and each pair $(x_i, f(x_i))$ is called an *example of f*. We call an example (x_i, a) a *positive example* if $a = 1$ and a *negative example* if $a = 0$. We call a the *label* of the example. The *sample space* of F, denoted S_F, is defined by $S_F = \{\mathrm{sam}(\bar{x}, f) : f \in F, \bar{x} \in X^t, t \geqslant 0\}$.

Let P be a probability distribution on X. For any $t \geqslant 1$, P^t denotes the corresponding product distribution on X^t. For any distribution T and random variable ψ, $E_T(\psi)$ denotes the expectation of ψ with respect to the distribution T.

2. Definitions and Main Results for Prediction Strategies

In this section we formally define prediction strategies and how we measure their performance. We then give examples, followed by upper and lower bounds on the performance of optimal prediction strategies.

[3] When $X = \mathbf{R}^n$, we make some measurability assumptions about F, as described in [BEHW89].

DEFINITION. A *deterministic prediction strategy* for F is a mapping $Q\colon S_F \times X \to \{0, 1\}$. For a deterministic prediction strategy, for any $f \in F$, $t \geqslant 1$, and $(x_1, ..., x_t) \in X^t$, let

$$\mathbf{M}^t_{Q,f}(x_1, ..., x_t) = \begin{cases} 1 & \text{if } \ Q(\operatorname{sam}((x_1, ..., x_{t-1}), f), x_t) \neq f(x_t) \\ 0 & \text{otherwise.} \end{cases}$$

Thus $\mathbf{M}^t_{Q,f}(x_1, ..., x_t)$ indicates whether or not the prediction strategy Q makes a mistake predicting the value of the concept f on x_t when given the value of f on $x_1, ..., x_{t-1}$.

We make similar definitions for randomized strategies.

DEFINITION. A *randomized prediction strategy* for F consists of a probability space Z with probability distribution P_Z together with a mapping $Q\colon S_F \times X \times Z \to \{0, 1\}$. The randomization is carried out by drawing a point at random from P_Z and giving this point as a parameter to Q. The mapping Q itself is deterministic. We let Q_r denote the deterministic strategy obtained by fixing the point from Z to be r, that is, $Q_r(s, x) = Q(s, x, r)$.

For a randomized strategy, let

$$\mathbf{M}^t_{Q,f}(x_1, ..., x_t) = \int \mathbf{M}^t_{Q_r,f}(x_1, ..., x_t)\, dP_Z(r).$$

This is the probability that the randomized strategy makes a mistake, given a fixed sequence $(x_1, ..., x_t)$. We assume that the random choice of r is independent of the random choices of the examples.

For both deterministic and randomized strategies, let $\hat{\mathbf{M}}_{Q,f}(t) = \sup E_{P^t}(\mathbf{M}^t_{Q,f})$, where the supremum is taken over all probability distributions P on X, and let $\hat{\mathbf{M}}_{Q,F}(t) = \sup \hat{\mathbf{M}}_{Q,f}(t)$, where the supremum is taken over all $f \in F$. This is the measure of predictive performance used in this paper. It represents the worst case probability of a mistake at the tth trial (worst case over the choice of f from F and over the choice of the distribution on X). In the case of randomized strategies, this follows from the independence of the randomization of the algorithm from the choice of examples.

The Permutation Mistake Bound

$\hat{\mathbf{M}}_{Q,F}(t)$ involves a supremum over P^t, where P is an arbitrary probability distribution on X; hence it is often difficult to measure. We deal with this by using a related bound $\hat{\hat{\mathbf{M}}}_{Q,F}(t)$ in whose definition drawing sequences according to P^t is replaced by drawing random permutations of fixed sequences of X^t. We will show that $\hat{\mathbf{M}}_{Q,F}(t) \leqslant \hat{\hat{\mathbf{M}}}_{Q,F}(t)$. The latter bound is easier to compute; we can estimate it using counting arguments.

DEFINITION. Let Γ_t denote the set of all permutations of $\{1, ..., t\}$. For a prediction strategy Q, and $f \in F$, let

$$\hat{\hat{\mathbf{M}}}_{Q,f}(t) = \sup \frac{1}{t!} \sum_{\sigma \in \Gamma_t} \mathbf{M}'_{Q,f}(x_{\sigma(1)}, ..., x_{\sigma(t)}),$$

where the supremum is taken over all $(x_1, ..., x_t) \in X^t$. Let $\hat{\hat{\mathbf{M}}}_{Q,F}(t) = \sup \hat{\hat{\mathbf{M}}}_{Q,f}(t)$ where the supremum is taken over all $f \in F$. $\hat{\hat{\mathbf{M}}}_{Q,F}(t)$ is called the *permutation mistake bound* of Q with respect to class F.

The following lemma can be used to show that $\hat{\mathbf{M}}_{Q,F}(t) \leqslant \hat{\hat{\mathbf{M}}}_{Q,F}(t)$.

LEMMA 2.1. *Let β be a real number, let Γ be a subset of Γ_t and let χ be a random variable defined on X^t for some $t \geqslant 1$. If for all sequences $(x_1, ..., x_t)$ of X^t,*

$$\frac{1}{|\Gamma|} \sum_{\sigma \in \Gamma} \chi(x_{\sigma(1)}, ..., x_{\sigma(t)}) \leqslant \beta,$$

then this implies that $E_{P^t}(\chi) \leqslant \beta$, for any distribution P on X.

Proof. For any permutation $\sigma \in \Gamma_t$

$$\int_{X^t} \chi(x_1, ..., x_t)\, dP^t(x_1, ..., x_t) = \int_{X^t} \chi(x_{\sigma(1)}, ..., \chi_{\sigma(t)})\, dP^t(x_1, ..., x_t).$$

Thus

$$\begin{aligned} E_{P^t}(\chi) &= \int_{X^t} \chi(x_1, ..., x_t)\, dP^t(x_1, ..., x_t) \\ &= \frac{1}{|\Gamma|} \sum_{\sigma \in \Gamma} \int_{X^t} \chi(x_{\sigma(1)}, ..., x_{\sigma(t)})\, dP^t(x_1, ..., x_t) \\ &= \int_{X^t} \frac{1}{|\Gamma|} \sum_{\sigma \in \Gamma} \chi(x_{\sigma(1)}, ..., x_{\sigma(t)})\, dP^t(x_1, ..., x_t) \\ &\leqslant \int_{X^t} \beta\, dP^t(x_1, ..., x_t) = \beta. \quad ▮ \end{aligned}$$

COROLLARY 2.1. *For all concept classes F, all prediction strategies Q for F, and all $t \geqslant 1$, $\hat{\mathbf{M}}_{Q,F}(t) \leqslant \hat{\hat{\mathbf{M}}}_{Q,F}(t)$.*

Proof. The above lemma applied with $\Gamma = \Gamma_t$ implies that $\hat{\mathbf{M}}_{Q,f}(t) \leqslant \hat{\hat{\mathbf{M}}}_{Q,f}(t)$ for every $f \in F$; the result follows from this. ▮

Note that this corollary would still hold if $\hat{\mathbf{M}}_{Q,F}(t)$ were defined with respect to any subset Γ of Γ_t.

It can still be quite difficult to calculate the permutation mistake bound in many cases. However, certain properties of the concept class F and the prediction strategy Q can be helpful in obtaining upper bounds on this quantity, which often turn out to be tight.

DEFINITION. Given a sample $s = (s_1, ..., s_t) \in S_F$, let $S = \{s_1, ..., s_t\}$. Let S^* denote the set of all finite sequences of elements of S. We call a set $B \subseteq S$ a Q-sufficient subsample of s if for every sample $s' \in S^*$ that contains B, after seeing s', Q predicts correctly on any further example from S, i.e., $Q(s', x) = a$ for all $(x, a) \in S$. In other words, B is a Q-sufficient subsample of s if, after any sequence of examples from s that contains all the examples in B (in any order and with any number of repetitions) has been seen, the hypothesis generated by Q is consistent with all examples in s. Let U denote the set of all elements of s that occur only once in s. If there exist any Q-sufficient subsamples of s then the Q-kernel of s is the intersection of U with the intersection of all Q-sufficient subsamples of s. Otherwise, the Q-kernel is undefined.

LEMMA 2.2. *Let Q be a deterministic prediction strategy, let f be a function from X into $\{0, 1\}$, and let $(x_1, ..., x_t) \in X^t$. Let $s = \operatorname{sam}((x_1, ..., x_t), f)$ and let K be the Q-kernel of s. Then*

$$\frac{1}{t!} \sum_{\sigma \in \Gamma_t} \mathbf{M}'_{Q,f}(x_{\sigma(1)}, ..., x_{\sigma(t)}) \leqslant \frac{|K|}{t}.$$

Proof. If $x_{\sigma(t)} \notin K$ then there is a Q-sufficient subsample of s, call it B, such that $B \subseteq \{x_{\sigma(1)}, ..., x_{\sigma(t-1)}\}$. In this case the prediction of Q on $x_{\sigma(t)}$ will be correct, since B is a Q-sufficient subsample of s. Thus Q cannot make a mistake unless $x_{\sigma(t)} \in K$. This occurs only in a fraction $|K|/t$ of the permutations, since each point in the kernel occurs only once in s. ∎

Examples: Rectangles, General Intersection Closed Classes, Unions of Intervals, and Halfspaces

The following examples illustrate how these ideas may be applied.

EXAMPLE 2.1. Let the domain $X = \mathbf{R}^n$ and let the concept class F_n consist of indicator functions of axis-parallel rectangular regions in $\mathbf{R}^n$. The positive examples of each concept in F_n form a single closed and bounded rectangular region. The boundaries are hyperplanes of dimension $n-1$ which are parallel to the coordinate axes.

We consider the following prediction strategy for this concept class. As long as no positive examples have been seen, the prediction strategy predicts 0 for each new point. When positive examples have been seen, the prediction strategy keeps track of the smallest axis-parallel closed rectangular region that contains all of the positive examples seen so far. It predicts 1 if the new point is contained in this region and 0 otherwise. The prediction strategy for indicator functions of intervals described in the introduction is a special case of this general strategy for $n = 1$. Call this general strategy Q.

It can be shown that for any sample s of any function in the concept class F_n, the Q-kernel of s contains at most $2n$ examples. We will demonstrate this for $n = 2$; the argument directly generalizes to higher dimensions. Consider any sequence s of points of $\mathbf{R}^2$ labeled according to some target rectangle in F_2. Since the prediction strategy Q always predicts using the hypothesis that is the smallest closed axis-parallel rectangle containing the positive examples seen so far, Q will never make a mistake on a negative example. Thus when s has no positive points, the Q-kernel of s is empty.

Assume that the sequence s contains at least one positive example. Let R denote the smallest closed axis-parallel rectangle containing the positive examples of the entire sequence s. Each edge of R will contain at least one point of s. Pick one such point from each edge of R to form a set B of size at most 4. Assume that Q is given the examples from s in any order with any number of repetitions. Once Q has seen all the examples in B, Q's hypothesis will be the indicator function of R, and this hypothesis will not change as further points from s are seen, and hence subsequent predictions of Q will be correct. Hence every set B obtained in this manner is a Q-sufficient subsample of s.

The Q-kernel K of s is contained in each Q-sufficient subsample of s. Since there exists at least one Q-sufficient subsample B of s with $|B| \leqslant 4$, $|K| \leqslant 4$. Clearly $|K| = 4$ only if there is only one positive point from s on each edge of R with each such point occurring only once in s, and no positive point is on two edges. Otherwise $|K| < 4$. These results easily generalize to n dimensions, for $n > 2$, giving $|K| \leqslant 2n$. It follows from Corollary 2.1 and Lemma 2.2 that $\hat{\mathbf{M}}_{Q,F_n}(t) \leqslant \hat{\hat{\mathbf{M}}}_{Q,F_n}(t) \leqslant 2n/t$; i.e., the probability that Q makes a mistake in predicting the value of the function f on the tth random example is at most $2n/t$ for any target function $f \in F_n$ and any distribution on $\mathbf{R}^n$.

Example 2.1 can be generalized quite a bit. The details are given in [HSW90], so we just sketch them briefly here.

EXAMPLE 2.2. Let F be a concept class on X. For any $G \subseteq F$, we define the conjunction of the concepts in G, denoted by $\bigcap G$, to be the concept on X that is 1 at just those points at which every concept in G is 1. For any set $S \subseteq X$, define the *closure* of S, denoted $CLOS(S)$, by $CLOS(S) = \bigcap \{f \in F: f(x) = 1 \text{ for all } x \in S\}$. For simplicity we define $\bigcap \emptyset = 0$ (the zero function). Thus for non-empty S, $CLOS(S) = 0$ if and only if there is no $f \in F$ such that $f(x) = 1$ for all $x \in S$. We say that F is *intersection closed* if $CLOS(S) \in F$ for all finite sets $S \subseteq X$ with $CLOS(S) \neq 0$.

Clearly F_n, as defined above, is intersection closed for each n. There are also many other instances of intersection closed concept classes in the learning literature. For example, the class of k-CNF Boolean functions on n variables studied in [Val84, Val85, KLPV87] and its generalizations studied in [H88a] are also intersection closed, as are various concept classes based on lattices and families of linear subspaces [HSW92, Shv88]. In the case where the domain X is finite, Natarajan has established an equivalence between intersection closed concept classes and those concept classes that can be PAC learned with no error on negative examples [N87] (see also [B88, Shv88]).

For intersection closed concept classes, a natural prediction strategy is to always predict using the hypothesis $CLOS(S)$, where S is the set of (points of) positive training examples seen so far. This is the (unique) maximally specific hypothesis in F that is consistent with the training examples. This prediction strategy is called the *closure algorithm*. The prediction strategy Q above for F_n is an instance of the closure algorithm.

If $T \subseteq S$ and $CLOS(T) = CLOS(S)$, then we say that T is a *spanning set* of S. This definition naturally extends to samples: if s is a sample of some target function in an intersection closed concept class F, S is the set of positive examples of s, and $B \subseteq S$ is a spanning set of S (ignoring the labels), then we say that B is a spanning set of s. It is clear that if Q is the closure algorithm, then B is a spanning set of s if and only if B is a Q-sufficient subsample of s. Hence, if all the points in s are distinct, the Q-kernel of s is the intersection of all the spanning sets of s.

In our rectangles example, we obtained a good bound on the performance of the closure algorithm by showing that every sample has a small spanning set, in particular a spanning set of size at most $2n$. This is not possible for all intersection closed concept classes. For example, if $X = \mathbf{R}^2$, F is the set of all indicator functions for convex subsets of X, and S is a finite subset of X, then $T \subseteq S$ is a spanning set of S if and only if T contains all the extremal points of S. Hence, even for arbitrarily large S, the smallest spanning set of S can be as large as the set S itself. However, as we will show, if F is intersection closed and has finite VC dimension, this cannot happen.

DEFINITION. Let X be a non-empty set and let F be a non-empty class of $\{0, 1\}$-valued functions on X. For any $S \subseteq X$, $\Pi_F(S)$ denotes the set of all $A \subseteq S$ such that there exists a function $f \in F$ that is 1 at all $x \in A$ and 0 at all $x \in S - A$. The *Vapnik–Chervonenkis dimension* of F, denoted VCdim(F), is

$$\sup\{|S| : S \subseteq X \text{ and } \Pi_F(S) = 2^S\}.$$

If F is empty then we say that VCdim(F) $= -1$.

It is easily verified that when F_n is the set of indicator functions for axis-parallel rectangles in $\mathbf{R}^n$, then VCdim(F_n) $= 2n$ (see [WD81] or [BEHW89]). Hence for functions in F_n, every sample s has a spanning set of size at most VCdim(F_n). This holds in general for intersection closed classes.

LEMMA 2.3. [N87, B88]. *If F is a (non-empty) intersection closed concept class on X and $S \subseteq X$ is a finite set with $CLOS(S) \neq 0$, then any minimal spanning set of S has size at most* VCdim(F).

Proof. Let T be a minimal spanning set of S. Then for every point $x \in T$, there is a concept $f_x \in F$ with $f_x(x) = 0$ but $f_x(y) = 1$ for all $y \in T - \{x\}$ (otherwise T is not minimal). Since $CLOS(S) \neq 0$, there is also a concept $f \in F$ such that $f(x) = 1$ for all $x \in T$. By taking intersections of these concepts, we can construct, for any $A \subseteq T$, a concept in F that is 1 on points in A and 0 on points in $T - A$. It follows that VCdim(F) $\geqslant |T|$. ▌

As in the rectangles example, using Corollary 2.1 and Lemma 2.2 we obtain the following result.

THEOREM 2.1 [HSW90]. *If F is intersection closed and Q is the closure algorithm, then* $\mathbf{M}_{Q,F}(t) \leqslant \hat{\mathbf{M}}_{Q,F}(t) \leqslant \mathrm{VCdim}(F)/t$.

Below we will construct a prediction strategy (usually not efficiently implementable) that obtains this performance bound on any concept class, whether or not it is intersection closed. However, before doing so, we will give two more examples of permutation mistake bounds for efficient prediction strategies. In the first example the concepts are defined in terms of unions.

EXAMPLE 2.3. Let the domain X be the real line and let F_n be the concept class containing the indicator function of each subset of X consisting of n disjoint closed intervals. Consider the following prediction strategy Q: if any of the previously seen points does not exceed the new point on which a prediction is to be made, make the prediction that matches the label of the greatest such point; otherwise, predict 0.

To describe the kernel, we order examples in increasing order according to where they fall on the real line. It is easy to see that for any sample s of any function $f \in F_n$, the Q-kernel of s contains the least example in s if and only if it is a positive example. In addition, the kernel also contains exactly those points that are labeled differently from their immediate predecessors. Clearly, for each interval of f, there are at most two points in the Q-kernel of s. It follows that the size of the Q-kernel is always at most $2n$. Using Corollary 2.1 and Lemma 2.2 we conclude that $\hat{\mathbf{M}}_{Q,F_n}(t) \leqslant \hat{\hat{\mathbf{M}}}_{Q,F_n}(t) \leqslant 2n/t$. It is easy to see that $\text{VCdim}(F_n) = 2n$.

In the last example we give a permutation mistake bound for a randomized prediction strategy. As in the previous example, the concept class is not intersection closed.

EXAMPLE 2.4. Let $X = \mathbf{R}^n$ and let F_n be the set of indicator functions of closed halfspaces of $\mathbf{R}^n$. It is well known that $\text{VCdim}(F_n) = n + 1$ [P84]. Given any finite sample s of a target function f in this class, there exists a hyperplane strictly separating the positive examples in s from the negative examples. Such a separating hyperplane can be found in polynomial time using linear programming (with the bit complexity model, using Karmarkar's algorithm [K84]). This gives a simple polynomial prediction strategy: given a sample $s \in S_{F_n}$ and $x \in \mathbf{R}^n$, solve a linear programming problem to obtain a separating hyperplane, and predict 1 if x lies in the halfspace determined by the hyperplane that contains the positive points of s, 0 otherwise. The exact strategy depends on how the linear-programming problem is constructed.

Not all strategies using linear programming to find a separating hyperplane have a useful permutation mistake bound. There exist such strategies for which the permutation mistake bound is 1, which is a trivial bound on $\hat{\mathbf{M}}$. It turns out that the problem is due primarily to the non-uniqueness of the optimal solution. Even in this case, a different analysis, not using the permutation mistake bound (see Theorem 4.1 below), gives a reasonable upper bound on $\hat{\mathbf{M}}_{Q,F}(t)$: for any prediction strategy Q whose hypothesis is always a halfspace consistent with all previous examples $\hat{\mathbf{M}}_{Q,F}(t) \leqslant (2(n+2)/t) \log 2(4et/n+1)$, for all $t > n+1$. With some care in the construction of the linear-programming algorithm, one can obtain a better bound than this by making use of the permutation mistake bound.

By randomizing the choice of the objective function of a suitable linear-programming problem one can arrange that with high probability the optimal solution is uniquely determined by a set of tight constraints whose size roughly equals the dimension of the problem. For an appropriately chosen linear-programming problem, this yields a prediction strategy LP with a permutation mistake bound of $(n+4)/t$, implying that $\hat{\mathbf{M}}_{LP,F_n}(t) \leqslant (n+4)/t$. Note that this bound on $\hat{\mathbf{M}}_{LP,F_n}(t)$ is essentially optimal.

As in Example 2.1, the permutation mistake bound depends on the number of tight constraints (due to sample points) at an extremal consistent hypothesis. The extremal hypothesis in this case is an optimal solution to a linear programming problem. As in the case of Example 2.1, it can be shown that we do not need to count redundant constraints in calculating the permutation mistake bound.

Prediction Strategies Based on the 1-Inclusion Graph

We present a randomized and a deterministic prediction strategy which use the 1-inclusion graph. Both are essentially optimal but the randomized strategy is slightly better. We first develop the background for the strategies.

DEFINITION. Given any sequence $\bar{x} = (x_1, ..., x_t) \in X^t$ and any concept class F over X, we construct a graph that we call the *1-inclusion graph for F with respect to* $\bar{x}$, denoted $G_F(\bar{x})$ [B72, AHW87].[4] The nodes of $G_F(\bar{x})$ are just the elements of $\Pi_F(\{x_1, ..., x_t\})$, as defined in conjunction with the VC dimension in Example 2.2 above. Let v and w be any two such nodes. They are connected with an edge if and only if the sets v and w differ by exactly one element x of $\bar{x}$ and x appears only once in $\bar{x}$. Each edge is labeled with the corresponding x.

Nodes of a 1-inclusion graph can have arbitrarily high degree, even for concept classes of Vapnik–Chervonenkis dimension 1. (Figure 1 gives an example for a class of VC dimension 2.) However, it turns out that the edges of any 1-inclusion graph for a concept class of Vapnik–Chervonenkis dimension d can be directed so that the outdegree of every node is at most d. This result, and a refinement of it that applies to randomized strategies, are the basis for the 1-inclusion graph prediction strategies. We develop these results in the next sequence of lemmas.

DEFINITION. The density of a graph G, denoted dens(G), is the number of its edges divided by the number of its nodes.

LEMMA 2.4. *Let F be a (nonempty) concept class and let G be any 1-inclusion graph for F. Then* $\text{dens}(G) < \text{VCdim}(F)$ *if* $\text{VCdim}(F) \geqslant 1$, *and if* $\text{VCdim}(F) = 0$ *then* $\text{dens}(G) = 0$ *as well.*

Proof. Let $d = \text{VCdim}(F)$. The case $d = 0$ is trivial. We assume that $d > 0$. Assume that $\bar{x} = (x_1, ..., x_t) \in X^t$ and $G = G_F(\bar{x})$. The proof proceeds by induction on t. When t is zero, G contains one node, and the result

[4] Our definition of the 1-inclusion graph reduces to the one given in these references when all the points in $\bar{x}$ are distinct.

follows trivially for all d. For the induction step, we will prove the result for $t > 0$, assuming that the result holds for $t-1$. There are two cases.

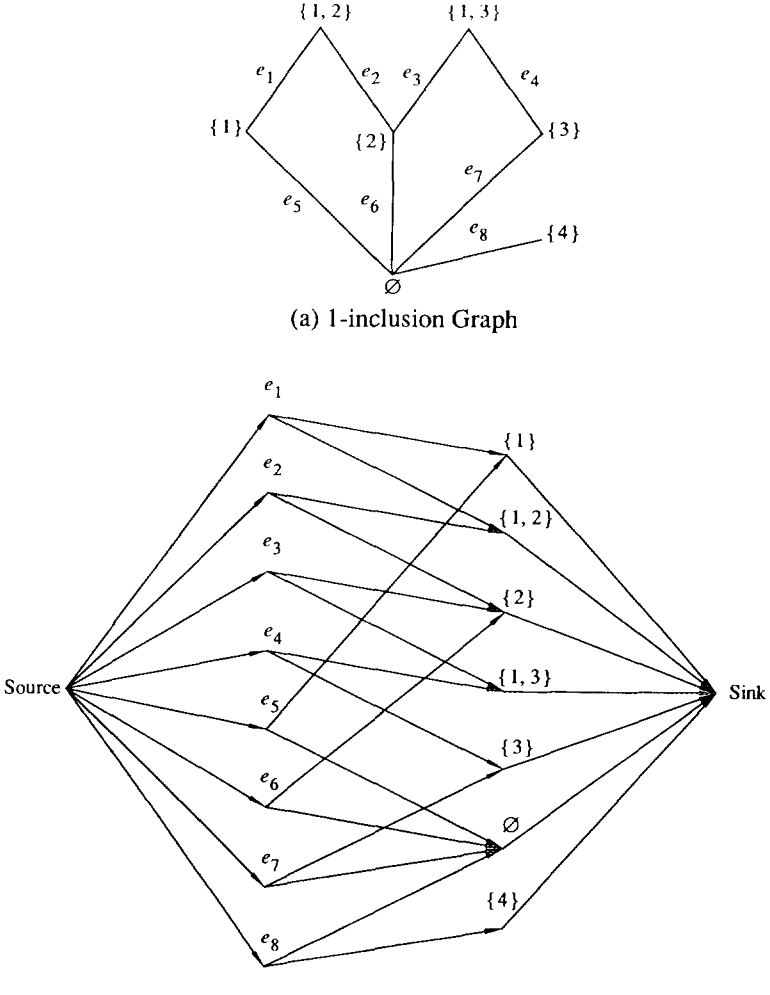
Case 1. $x_t = x_i$ for some $1 \leqslant i < t$. Let $H = G_F(x_1, ..., x_{t-1})$. Since x_t already appears in $\{x_1, ..., x_{t-1}\}$, the node set of H is the same as that of G, and the edge set of G is the same as the edge set of H, except that all edges labeled with x_t are missing. Hence $\mathrm{dens}(G) \leqslant \mathrm{dens}(H) < d$ by assumption.

Case 2. $x_t \neq x_i$ for all $1 \leqslant i < t$. We construct a mapping η of the nodes of G onto the nodes of another 1-inclusion graph H. Let $\bar{y} = (x_1, ..., x_{t-1})$. Then $H = G_F(\bar{y})$. Note that nodes of G are subsets of $\{x_1, ..., x_t\}$ and nodes of H are subsets of $\{x_1, ..., x_{t-1}\}$. The mapping η from nodes of G to nodes of H is defined by $\eta(v) = v - \{x_t\}$. Thus two nodes of G map to the same node of H if they differ only in whether or not they contain x_t. Note that either one or two nodes of G map to each node of H. Now let W be the set of all nodes of H whose inverse image under η contains two nodes of G. If W is empty, then the number of nodes of G equals the number of nodes of H and for every edge of G there exists a distinct corresponding edge in H. Thus in this case $\mathrm{dens}(G) \leqslant \mathrm{dens}(H) < d$ by the induction hypothesis, and we are done.

Otherwise, let F' be the concept class over X whose concepts are the indicator functions of the subsets of X which are the elements of W. One can show that the Vapnik–Chervonenkis dimension of F' is at most $d-1$ (if not, then there exists an $S \subseteq \{x_1, ..., x_{t-1}\}$ with $|S| = d$ such that $\Pi_F(S \cup \{x_t\}) = 2^{S \cup \{x_t\}}$, contradicting the fact that $\mathrm{VCdim}(F) = d$.) We consider a third 1-inclusion graph, $J = G_{F'}(\bar{y})$. The nodes of J are just the elements of W. Let N_G, N_H, and N_J denote the number of nodes of G, H, and J, respectively. Let E_G, E_H, and E_J denote the number of edges of G, H, and J, respectively. The number of nodes of G is just the number of nodes of H plus the number of nodes of H to which two nodes of G map under η; i.e., $N_G = N_H + N_J$. To count the edges of G, first note that there will be one edge in G for each element of W, since any two nodes which map to the same node of H will be joined by an edge. Call these edges edges of the first type. The number of edges in G of the first type will equal N_J. For each other edge of G, there must be distinct nodes u and v of H such that the edge joins a node in $\eta^{-1}(\{u\})$ to a node of $\eta^{-1}(\{v\})$. Such edges can only occur when u and v are connected by an edge in H. There can be two such edges for a given u and v only if u and v are both in W, and then u and v will also be connected by an edge in J. Thus the number of edges in G of the second type will be bounded by the number of edges of H plus the number of edges of J. Thus we have $E_G \leqslant E_H + E_J + N_J$. By the induction hypothesis, $E_H < dN_H$. Also, $E_J \leqslant (d-1) N_J$; this inequality follows from the induction hypothesis for $d > 1$. For $d = 1$, it follows from the fact that in that case $E_J = 0$. Thus

$$\operatorname{dens}(G) \leqslant \frac{E_H + E_J + N_J}{N_H + N_J} < \frac{dN_H + (d-1)\,N_J + N_J}{N_H + N_J} = d. \quad \blacksquare$$

An alternate proof of the above lemma is given in [H95]. It should be noted that while the inequality in the above result is strict, it is still tight



(a) 1-inclusion Graph

(b) Corresponding Network

FIG. 2. Constructing a network from a 1-inclusion graph.

in the sense that there are cases where dens(G) gets arbitrarily close to VCdim(F) for all VC dimensions greater than zero. (For example, for VC dimension d, consider the class over some finite domain that consists of all functions that are 1 on exactly d points, and consider the 1-inclusion graph with respect to all but 1 of the points in the domain. If the domain has size at least $2d$ then this class has VC dimension d, and as the domain size grows, it is easy to see that the density of the described 1-inclusion graph grows arbitrarily close to d.)

We now devise a method for assigning probabilities to the ends of each edge of a 1-inclusion graph. The probabilities assigned to the two ends of an edge sum to one and these probabilities are used to determine a randomized prediction. Let us fix a concept class F of VC-dimension $d > 0$ and a sequence of points $\bar{x} = (x_1, ..., x_t) \in X^t$. (If $d = 0$ then the class has only one concept, and a trivial prediction algorithm suffices.) Consider the 1-inclusion graph $G_F(\bar{x})$ for the concept class F on the sequence of points $\bar{x}$. A probability will be associated with each end of each edge of this graph. We determine these probabilities by solving an associated network-flow problem. The following densities (number of edges over number of nodes) will be used in defining the capacities used in the network.

DEFINITION. For any node v of $G_F(\bar{x})$, let $MD_F(\bar{x})$ denote the maximum density of any subgraph of $G_F(\bar{x})$.

For example, the density of the 1-inclusion graph of Fig. 2a is 8/7, whereas its maximum density is 7/6.

We construct a network as follows: There is a single source node and a single sink. The flow from the source to the sink passes through two intermediate layers of nodes (see Fig. 2b).

The first intermediate layer has one node corresponding to every edge in the 1-inclusion graph, and the second layer has one node corresponding to every node in the 1-inclusion graph. The network edges and their capacities are as follows: There is a network edge directed from the source to each node of the first intermediate layer. Each such edge has capacity 1. Each network node n_1 of the first layer corresponds to some edge e of the 1-inclusion graph; this edge e joins two nodes, v_1 and v_2, of the 1-inclusion graph. These nodes, in turn, correspond to two nodes n_2 and n_3 in the second intermediate network layer. Edges are directed out from network node n_1 to nodes n_2 and n_3; this construction is carried out for each node in the first intermediate layer. Each of the edges joining the first to the second layer has infinite capacity (or, equivalently for our purposes, any capacity greater than or equal to 1). Finally, there is a network edge directed from each node of the second intermediate layer to the sink with capacity $MD_F(\bar{x})$. Denote the described network by $N_F(\bar{x})$. We also define a slightly modified network $I_F(\bar{x})$ in which all capacities are integral: $I_F(\bar{x})$ equals $N_F(\bar{x})$ except that the capacities of the edges from the second intermediate layer to the sink are $\lceil MD_F(\bar{x}) \rceil$.

LEMMA 2.5. *For any network $N_F(\bar{x})$ its maximum flow is the number of edges in $G_F(\bar{x})$. For any network $I_F(\bar{x})$ there is an integral flow equal to the number of edges in $G_F(\bar{x})$.*

Proof. The second part clearly follows from the first part: any flow of $N_F(\bar{x})$ is also a flow of $I_F(\bar{x})$ and since all capacities of $I_F(\bar{x})$ are integral there exists a maximum flow of $I_F(\bar{x})$ that is integral [W86]. For the first part, let m be the number of edges in $G_F(\bar{x})$. Since in $N_F(\bar{x})$ there are m edges of capacity 1 from the source, the maximum flow is at most m. To show that it is at least m, we use the fact that the maximum flow equals the capacity of the minimum cut (set of edges) separating all paths from the source to the sink. Fix attention on a minimum cut. Since the capacity of the minimum cut is finite, it does not contain any network edge between the first and second intermediate layers. If the minimum cut does not contain any edges from the second layer to the sink then its capacity must be m since it must consist of all edges connected to the source. In this case we are done. Otherwise, consider those edges from the second layer to the sink that are cut. Let $\tilde{W}$ denote the set of network nodes in the second intermediate layer that are incident to these edges, and let W be the set of corresponding 1-inclusion graph nodes. We will now find an upper bound on the number of network edges from the source to the first intermediate layer that are not cut. Consider any such edge that is not cut. This edge terminates at some network node e that corresponds to a 1-inclusion graph edge. We claim that both endpoints of this 1-inclusion graph edge are in W. Otherwise, there will be a network edge from e to a network node not in $\tilde{W}$, and there will be an uncut path from the source to the sink passing through that edge. Thus e corresponds to an edge in the subgraph G' of the 1-inclusion graph induced by W. It follows that the number of uncut edges from the source to the first intermediate layer is bounded by the number of edges in G' which equals $D' \, |W|$, where D' is the density of the subgraph G'. The number of cut edges from the source is therefore at least $m - D' \, |W|$. Since each of these edges has capacity 1, and since each cut edge from a node of $\tilde{W}$ to the sink has capacity $MD_F(\bar{x}) \geqslant D'$, the total capacity of the minimum cut is at least m. ∎

Remark. It is easy to see that $MD_F(\bar{x})$ is the minimum capacity for the edges connecting the second intermediate layer to the sink that allows the maximum flow of the network $N_F(\bar{x})$ be equal to the number of edges of $G_F(\bar{x})$. If H is a subgraph of $G_F(\bar{x})$ of maximum density, then since each edge of H is responsible for one unit of flow, the total capacity of all edges in the network connecting the nodes corresponding to the vertices of H to the sink must be at least $MD_F(\bar{x})\,|H|$. We will use this property later to search for the value of $MD_F(\bar{x})$.

We next describe the randomized prediction strategy. Assume that we have some fixed ordering on the elements of X, and let the sequence $\bar{x}_s$ denote the set $\{x_i : 1 \leqslant i \leqslant t\}$ in sorted order. Use any standard deterministic network-flow algorithm to find a maximum flow for the network $N_F(\bar{x}_s)$. (The produced maximum flow does not depend on the order of the points in $\bar{x}$.) Denote the sets of nodes and edges of the 1-inclusion graph by V and E, respectively. Construct a map $p: V \times E \rightarrow [0, 1]$ as follows: For each edge $e \in E$ with endpoints v_1 and v_2, there are network nodes $\tilde{e}$, $\tilde{v}_1$, and $\tilde{v}_2$ corresponding to e, v_1, and v_2, respectively, and there are network edges joining the node $\tilde{e}$ to $\tilde{v}_1$ and $\tilde{v}_2$. Let u_1 and u_2 denote the flows on the edges to $\tilde{v}_1$ and $\tilde{v}_2$, respectively. Since the flow into $\tilde{e}$ is 1, $u_1 + u_2 = 1$. We let $p(v_1, e) = u_1$ and $p(v_2, e) = u_2$. For every edge e, for every node v not incident with e, we let $p(v, e) = 0$. Note that for any 1-inclusion graph node v, the flow into the corresponding network node is $\sum_{e \in E} p(v, e)$. Since the flow out of the network node v in $N_F(\bar{x}_s)$ is at most $MD_F(\bar{x})$, we have for every $v \in V$ that

$$\sum_{e \in E} p(v, e) \leqslant MD_F(\bar{x}). \tag{1}$$

The randomized strategy uses the constructed flow of $N_F(\bar{x}_s)$ to make randomized predictions. Each node of the 1-inclusion graph is an element of $\Pi_F(\{x_1, ..., x_t\})$. The labeling of $x_1, ..., x_{t-1}$ in the sample can be consistent with at most two of the nodes, one for each possible labeling of x_t. There must be at least one consistent node since the sample is consistent with some concept in F. Also, if $x_t = x_j$, for $1 \leqslant j < t$, then there is always exactly one consistent node. In the case when there is one consistent node v, the strategy makes whichever prediction is consistent with that node. If there are two consistent nodes, v_1 and v_2, then they differ only for x_t. Thus there must be an edge e joining the nodes. The strategy chooses node v_1 with probability $p(v_2, e)$ and node v_2 with probability $p(v_1, e)$. As noted earlier, these probabilities sum to 1. (In terms of our formal definition of randomized strategy, one way to obtain this randomization is by letting $Z = [0, 1]$ with the uniform distribution. The algorithm receives a parameter r chosen uniformly at random from Z and then chooses node v_1 if $r \leqslant p(v_2, e)$.) It then makes a prediction consistent with the chosen node.

For the deterministic prediction strategy a maximum integral flow [W86] for the network $I_F(\bar{x}_s)$ is constructed. The edges connecting the source to the first layer all have flow one. Since the flow is integral the mapping p defined above has the following property: for each edge $e \in E$

with endpoints v_1 and v_2, one of $\{p(v_1, e), p(v_2, e)\}$ is one and the other zero. So the prediction based on this mapping is deterministic. Also, since the flow out of the network node v in $I_F(\bar{x}_s)$ is at most $\lceil MD_F(\bar{x})\rceil$, we have for every $v \in V$ that

$$\sum_{e \in E} p(v, e) \leqslant \lceil MD_F(\bar{x})\rceil. \tag{2}$$

If we direct the edges of the 1-inclusion graph so that each edge e is directed towards its endpoint v for which $p(v, e) = 0$, then we arrive at another description of the deterministic strategy. When a prediction is to be made and there are two consistent nodes, the strategy examines the edge joining the nodes, and makes a prediction consistent with the node towards which that edge is directed. Using this method of directing the edges, the following theorem follows immediately from inequality (1) and Lemma 2.4:

THEOREM 2.2. *For any concept class F on X and any $\bar{x}$ over X, the edges of the 1-inclusion graph $G_F(\bar{x})$ can be directed so that the out-degree of every node is at most* $\lceil MD_F(\bar{x})\rceil \leqslant \mathrm{VCdim}(F)$.

An alternate way of proving this result can be obtained using results in [AT92]. The above prediction strategies lead to the bounds given below in Theorem 2.3, which makes use of the following definition.

DEFINITION. For any $t \geqslant 1$ and concept class F on X, let $\mathrm{maxdens}_t(F)$ be the maximum[5] of $MD_F(\bar{x})$ over all $\bar{x} \in X^t$.

THEOREM 2.3. (i) *For any concept class F there exists a randomized prediction strategy Q such that for any $f \in F$ and any $\bar{x} \in X^t$*

$$\frac{1}{t!}\sum_{\sigma \in \Gamma_t} \mathbf{M}'_{Q,f}(x_{\sigma(1)}, ..., x_{\sigma(t)}) \leqslant \frac{MD_F(\bar{x})}{t}$$

(i.e., for any sequence of points $\bar{x} = (x_1, ..., x_t)$ and any target concept $f \in F$, the expected fraction of all permutations of $\bar{x}$ for which the strategy Q makes a mistake in predicting the value of f on the last point, given the value of f on the previous points, is at most $MD_F(\bar{x})/t$), and hence the randomized strategy Q has permutation mistake bound $\hat{\mathbf{M}}_{Q,F}(t)$ (which upper bounds $\hat{\mathbf{M}}_{Q,F}(t)$) of at most $\mathrm{maxdens}_t(F)/t < \mathrm{VCdim}(F)/t$. *(The strict inequality fails to hold in the trivial case that* $\mathrm{VCdim}(F) = 0$.)

[5] Note that the maximum is well defined since there are only finitely many distinct values that $MD_F(\bar{x})$ can take for a fixed t.

(ii) *For any concept class F there exists a deterministic prediction strategy Q such that for any $f \in F$ and any $\bar{x} \in X^t$,*

$$\frac{1}{t!} \sum_{\sigma \in \Gamma_t} \mathbf{M}'_{Q,f}(x_{\sigma(1)}, ..., x_{\sigma(t)}) \leqslant \frac{\lceil MD_F(\bar{x}) \rceil}{t}$$

and[6] $\hat{\mathbf{M}}_{Q,F}(t) \leqslant \hat{\hat{\mathbf{M}}}_{Q,F}(t) \leqslant \lceil \text{maxdens}_t(F) \rceil / t \leqslant \text{VCdim}(F)/t$.

Proof. Let $\bar{x} = (x_1, ..., x_t) \in X^t$ and let V and E be the sets of nodes and edges of the 1-inclusion graph $G_F(\bar{x})$. Suppose that node v of the 1-inclusion graph corresponds to the equivalence class containing the target function f. Suppose that the randomized prediction strategy has seen $\text{sam}((x_1, ..., x_{t-1}), f)$ and a prediction is desired for x_t. Then if there is an edge e labeled x_t that is incident with v then the probability the randomized algorithm makes a mistake is $p(v, e)$; otherwise, the probability of a mistake is 0. Thus $\mathbf{M}'_{Q,f}(x_1, ..., x_t) \leqslant p(v, e)$, and hence

$$\frac{1}{t!} \sum_{\sigma \in \Gamma_t} \mathbf{M}'_{Q,f}(x_{\sigma(1)}, ..., x_{\sigma(t)}) \leqslant \frac{\sum_{e \in E} p(v, e)}{t}.$$

By inequality (1) and the definition of $MD_F(\bar{x})$, the above is upper bounded by $MD_F(\bar{x})/t$. Thus $\hat{\hat{\mathbf{M}}}_{Q,f}(t) \leqslant MD_F(\bar{x})/t \leqslant \text{maxdens}_t(F)/t$. Finally, by application of Lemma 2.4 to the induced subgraphs of the 1-inclusion graph $G_F(\bar{x})$, it follows that $\text{maxdens}_t(F) < \text{VCdim}(F)$, whenever $\text{VCdim}(F) > 0$. This completes the proof of part (i) for the randomized strategy.

The proof of the bounds for the deterministic strategy is essentially the same except that inequality (2) is used in place of (1). ▌

Efficiency of the 1-Inclusion Graph Prediction Strategy

The deterministic and randomized 1-inclusion graph prediction strategies may not yield efficient algorithms. However, if the VC dimension of the concept class is small, they may be computationally feasible.

DEFINITION. A concept class F is *polynomially recognizable* if there exists a polynomial algorithm that, given a sample, decides whether there is a concept in F consistent with the sample.

THEOREM 2.4. *If F is has finite* VC *dimension and is polynomially recognizable then the 1-inclusion graph can be created and applied to make deterministic or randomized predictions in time polynomial in the size of the sample.*

[6] The upper bound previously obtained [HLW90] was 2 VCdim$(F)/t$.

Proof Sketch. A theorem of Sauer [Sau72] shows that for any $(x_1, ..., x_t) \in X^t$, the cardinality of $\Pi_F\{x_1, ..., x_t\}$ is $O(t^{\mathrm{VCdim}(F)})$. Hence by Lemma 2.4 the size (number of edges plus number of vertices) of the 1-inclusion graph $G_F(\bar{x})$ is $O(\mathrm{VCdim}(F)\, t^{\mathrm{VCdim}(F)})$. If VCdim($F$) is finite and one can tell in polynomial time whether or not there is any function in F that is consistent with a given sequence of examples, then Lemma 3.2.2 of [BEHW89] shows how the sets in $\Pi_F(\bar{x})$ can be listed in polynomial time. Given this list, it is easy to construct $G_F(\bar{x})$ in polynomial time. To construct $N_F(\bar{x}_s)$ we need to search[7] for the value $MD_F(\bar{x})$. This can be done in polynomial time using a standard network flow algorithm [W86] since $MD_F(\bar{x})$ can have only polynomially many different values and $MD_F(\bar{x})$ is the minimum capacity of the edges connecting the second layer to the sink that allows the maximum flow of $N_F(\bar{x}_s)$ to be equal to the number of edges of $G_F(\bar{x})$. Thus maximum flows for $N_F(\bar{x}_s)$ and $I_F(\bar{x}_s)$ can be constructed in polynomial time. It is easy to predict based on these maximum flows. ▌

Cumulative Mistake Bounds

A more general way to analyze prediction strategies is to consider the expected total number of mistakes in a sequence of trials from some time t' up to time t, instead of just the probability of mistake at time t. Weighted averages could also be considered. The total number of mistakes beginning from the first example has been studied extensively in the model where the points in X are selected by an adversary, rather than being drawn independently at random [L87, B90, F89, GRS93, HSW90, 92, L89a, b, MT92, MT94a, b]. However, for continuous domains, even for simple concept classes such as indicator functions for intervals on the real line, the best prediction strategies will make a mistake on every trial for some sequences of points. Thus it is also useful to consider the model in which the points are drawn independently at random from some distribution on X, and the expected total number of mistakes is measured. Here again it may be the case that this is infinite, and hence we may wish to study the expected number of mistakes made during the first t trials as a function of t. Such measures are studied in [HSW90]. We will call them *cumulative mistake bounds.*

Since our upper bounds on the probability of mistake at trial t hold for any distribution on the domain X, using the linearity of the expectation we can simply sum these bounds to obtain upper bounds on the expected total number of mistakes.

[7] If one is willing to accept the weaker bounds expressed in terms of the VC dimension that are given in Theorem 2.3, then one can avoid this search by setting the capacities of the edges incident to the sink to VCdim(F).

COROLLARY 2.2. *Assume $1 \leqslant t' \leqslant t$. For any concept class F there exists a prediction strategy Q such that for any target function f in F and any distribution on the domain of f, if t examples of f are drawn independently at random and given to Q, then the expected total number of mistakes in prediction on examples indexed t' to t is at most*

$$\sum_{i=t'}^{t} \frac{\mathrm{VCdim}(F)}{i}$$

In particular, if $t' = 1$ then the expected total number of mistakes is less than $(\ln(t)+1)\,\mathrm{VCdim}(F)$, and if $t' = \lceil t/2 \rceil + 1$ then the expected total number of mistakes is less than $\ln(2)\,\mathrm{VCdim}(F)$.

Proof. This follows directly from Theorem 2.3 and well known bounds on partial sums of the harmonic series. ▌

3. LOWER BOUNDS

We now show that the bounds on $\hat{\mathbf{M}}_{Q,F}(t)$ for both the deterministic and randomized algorithms are tight to within a constant factor.

DEFINITION. A (non-empty) concept class $F \subseteq 2^X$ is *trivial* if F contains exactly one concept, or if F consists of two functions $f, g: X \to \{0, 1\}$ such that $f = 1 - g$.

THEOREM 3.1. *Assume that F is non-trivial and Q is any prediction strategy. (Q can be deterministic or randomized.)*

(i) *If $\mathrm{VCdim}(F)$ is infinite then $\hat{\mathbf{M}}_{Q,F}(t) \geqslant \frac{1}{2}$ for all $t \geqslant 1$.*

(ii) *If $\mathrm{VCdim}(F)$ is finite then for all $t > \mathrm{VCdim}(F)$, $\hat{\mathbf{M}}_{Q,F}(t) \geqslant \max(1, \mathrm{VCdim}(F) - 1)/2et$, where e is the base of the natural logarithm.*

Proof. We use the technique from [EHKV89]. Here we give only the proof of part (ii); part (i) is similar. We first consider the case $\mathrm{VCdim}(F) \geqslant 2$. Let $k = \mathrm{VCdim}(F) - 1$ and fix $t \geqslant k$. Assume $X_0 = \{y_0, ..., y_k\} \subseteq X$ is shattered by F. Let P be the distribution on X defined by

$$P(y_i) = \frac{1}{t}, \qquad 1 \leqslant i \leqslant k,$$

$$P(y_0) = 1 - \frac{k}{t},$$

and

$$P(x)=0, \qquad x\in X-X_0.$$

Let $p=P^t\{(x_1,...,x_t)\in X_0^t: x_t\neq x_i, 1\leqslant i\leqslant t\}$. Note that $p\geqslant P^t\{(x_1,...,x_t)\in X_0^t: x_t\neq y_0$ and $x_t\neq x_i, 1\leqslant i<t\}=(k/t)(1-1/t)^{t-1}\geqslant k/et$.

Let $F_0\subseteq F$ be a set consisting of $2^{|X_0|}$ functions such that $\Pi_{F_0}(X_0)=\Pi_F(X_0)=2^{X_0}$ and let T be the uniform distribution on F_0. Let $\mathbf{M}_Q^t(\bar{x},f)=\mathbf{M}_{Q,f}^t(\bar{x})$ for all $f\in F_0$ and $\bar{x}\in X_0^t$. Given any sample $s=((x_1,a_1),...,(x_{t-1},a_{t-1}))\in S_{F_0}$ and $x_t\in X_0$ such that $x_t\neq x_i$, $1\leqslant i<t$, half of the functions in F_0 that are consistent with s are 1 at x_t and the other half are 0 at x_t. Thus for any fixed $\bar{x}=(x_1,...,x_t)\in X_0^t$ such that $x_t\neq x_i$, $1\leqslant i<t$, $E_T(\mathbf{M}_Q^t(\bar{x},\cdot))=1/2$ for any prediction strategy Q. This implies that $E_{P^t\times T}(\mathbf{M}_Q^t)\geqslant p/2\geqslant k/2et$. Hence there exists $f_0\in F_0$ such that $E_{P^t}(\mathbf{M}_Q^t(\bar{x},f_0))\geqslant k/2et$.

For VCdim$(F)\geqslant 2$, the above argument shows that for all $t\geqslant$ VCdim$(F)-1$ there exists $f\in F$ such that $E_{P^t}(\mathbf{M}_{Q,f}^t)\geqslant(\text{VCdim}(F)-1)/2et$. If VCdim$(F)<2$, then we can use the fact that F is non-trivial to find concepts f_1 and f_2 in F and points a and b in X such that a is in f_1 but not in f_2 and b is either both in f_1 and in f_2 or neither in f_1 nor in f_2. We then set $P(a)=1/t$ and $P(b)=1-1/t$. The remainder of the proof is similar to the above argument and shows that $E_{P^t}(\mathbf{M}_{Q,f}^t)\geqslant 1/2et$ for either $f=f_1$ or $f=f_2$. ∎

The above lower bound on the probability of a mistake at the tth trial is obtained by constructing for each t a distribution $P(t)$ for which we show that $E_{P(t)^t}(\mathbf{M}_{Q,f}^t)$ exceeds the stated lower bound for some $f\in F$. Because the distribution is changed with t, we do not obtain a lower bound for the expected total number of mistakes made in a sequence of trials by adding the lower bounds for the separate trials. For certain concept classes, we are able to derive lower bounds by a different approach, using a single distribution on the domain for all t, thus yielding lower bounds on the expected total number of mistakes made in a sequence of trials as well as on the probability of a mistake in a single trial.

We first consider the domain $X_d=[0,1]\times\{1,...,d\}$ and the concept class $G_d=\{f_{\bar{q}}:\bar{q}\in[0,1)^d\}$, where $f_{(q_1,...,q_d)}$ denotes the function from X_d to $\{0,1\}$ defined by $f_{(q_1,...,q_d)}(\langle x,j\rangle)=1$ if and only if $x\leqslant q_j$. We will refer to concept class G_d as the *class of unions of d initial segments.*

In the next lower bound, as in the proof of the previous lower bound, we will bound how well any prediction strategy can perform on a random concept $f\in G_d$ drawn w.r.t. some distribution U on G_d. As in that proof, let $\mathbf{M}_Q^t$ be a new random variable on the domain $X_d^t\times G_d$ such that

$$\mathbf{M}_Q^t(x_1,...,x_t,f)=\mathbf{M}_{Q,f}^t(x_1,...,x_t).$$

The *uniform* distribution U on G_d picks each q_j independently and uniformly in the interval $[0, 1) \times \{j\}$. The uniform distribution P on X_d chooses with probability $1/d$ a particular j and then a point is drawn uniformly in the interval $[0, 1] \times \{j\}$. Note that in the following lower bound the distribution P on X_d does not change with t as was the case in Theorem 3.1.

THEOREM 3.2. *Let G_d be the concept class of d initial segments on the domain X_d. For all $d \geqslant 1$, for the uniform distributions U on G_d and P on X_d, for any prediction strategy Q for G_d, and for any $t \geqslant 1$,*

$$\hat{\mathbf{M}}_{Q,G_d}(t) \geqslant E_{P^t \times U}(\mathbf{M}^t_Q) \geqslant \frac{d}{2t} + \frac{d^2}{2t(t+1)}\left(\left(1 - \frac{1}{d}\right)^{t+1} - 1\right).$$

Proof. The first inequality follows directly from the definition of $\hat{\mathbf{M}}_{Q,G_d}(t)$. For the second inequality observe that

$$\begin{aligned} E_{P^t \times U}(\mathbf{M}^t_Q) &= \int_{X^t_d \in G_d} \mathbf{M}^t_Q \,\mathbf{d}P^t \times U \\ &= \int_{X^t_d}\left(\int_{G_d} \mathbf{M}^t_Q(x_1, ..., x_t, f)\,\mathbf{d}U(f)\right)\mathbf{d}P^t(x_1, ..., x_t), \qquad (3) \end{aligned}$$

by Fubini's Theorem. Temporarily fixing some $x_1, ..., x_t$, we next determine a lower bound on $\int_{G_d} \mathbf{M}^t_Q(x_1, ..., x_t, f)\,\mathbf{d}U(f)$. Assume that $x_t = \langle p, k\rangle$. Let $\lambda = \max\{x : x \leqslant p$ and $\langle x, k\rangle \in \{x_1, ..., x_{t-1}\} \cup \{\langle 0, k\rangle\}\}$ and let $\rho = \min\{x : x \geqslant p$ and $\langle x, k\rangle \in \{x_1, ..., x_{t-1}\} \cup \{\langle 1, k\rangle\}\}$. Let $L = \{f_{(q_1, ..., q_d)} \in G_d : \lambda \leqslant q_k < p\}$ and let $R = \{f_{(q_1, ..., q_d)} \in G_d : p \leqslant q_k < \rho\}$. Because the prediction of the strategy depends only on the sample it has seen (and possibly also on independent randomization), we have $\mathbf{M}^t_Q(x_1, ..., x_t, f) = 1 - \mathbf{M}^t_Q(x_1, ..., x_t, g)$ whenever $f \in L$ and $g \in R$ satisfy $f(\langle x, j\rangle) = g(\langle x, j\rangle)$ for all $\langle x, j\rangle \in G_d$ such that $j \neq k$. (This holds since all of the labels that the prediction strategy has seen before its prediction match for the two concepts f and g, but the correct prediction for x_t will be different in the two cases.) Thus if we let $\tau(x_1, ..., x_t) = \min(p - \lambda, \rho - p)$, then

$$\begin{aligned} &\int_{G_d} \mathbf{M}^t_Q(x_1, ..., x_t, f)\,\mathbf{d}U(f) \\ &\qquad \geqslant \int_L \mathbf{M}^t_Q(x_1, ..., x_t, f)\,\mathbf{d}U(f) + \int_R \mathbf{M}^t_Q(x_1, ..., x_t, f)\,\mathbf{d}U(f) \\ &\qquad \geqslant \tau(x_1, ..., x_t). \end{aligned}$$

The final inequality above is obtained by rewriting the integrals as iterated integrals with respect to the variables $q_1, ..., q_d$, where the innermost integral is with respect to q_k. It is then straightforward to combine the integrals to obtain this inequality. Substituting into (3) we get

$$E_{P^t \times U^t}(\mathbf{M}'_Q) \geqslant \int_{X_d^t} \tau(x_1, ..., x_t)\, \mathbf{d}P^t(x_1, ..., x_t)$$
$$= E_{P^t}(\tau) = \int_0^{1/2} uF'(u)\, \mathbf{d}u,$$

where F is the cumulative distribution of τ, i.e., $F(u) = P^t\{\tau \leqslant u\} = 1 - P^t\{\tau > u\}$. Let $I_{\tau > u}(x_1, ..., x_t)$ be the function that is 1 if $\tau(x_1, ..., x_t) > u$, 0 otherwise. Then

$$P^t\{\tau > u\} = \int_{X_d^t} I_{\tau > u}(x_1, ..., x_t)\, \mathbf{d}P^t(x_1, ..., x_t)$$
$$= \int_{X_d} \left(\int_{X_d^{t-1}} I_{\tau > u}(x_1, ..., x_t)\, \mathbf{d}P^{t-1}(x_1, ..., x_{t-1}) \right) \mathbf{d}P(x_t)$$
$$= \int_{W} \left(\int_{X_d^{t-1}} I_{\tau > u}(x_1, ..., x_t)\, \mathbf{d}P^{t-1}(x_1, ..., x_{t-1}) \right) \mathbf{d}P(x_t),$$

where W denotes the area $\bigcup_{j=1}^{d} (u, 1-u) \times \{j\}$. Note that $I_{\tau > u}(x_1, ..., x_t) = 1$ if (i) $x_t \in W$ and (ii) the sequence $(x_1, ..., x_t)$ is in $(X_d - [p-u, p+u] \times \{k\})^{t-1}$, where $x_t = (p, k)$. Since $P(X_d - [p-u, p+u] \times \{k\}) = 1 - 2u/d$, (ii) occurs with probability $(1 - 2u/d)^{t-1}$ and hence

$$P^t\{\tau > u\} = \int_W \left(1 - \frac{2u}{d}\right)^{t-1} \mathbf{d}(P(x_t)) = \left(1 - \frac{2u}{d}\right)^{t-1} P(W)$$
$$= \left(1 - \frac{2u}{d}\right)^{t-1} (1 - 2u).$$

It follows that $F(u) = 1 - (1 - 2u/d)^{t-1}(1 - 2u)$ for $0 \leqslant u \leqslant \frac{1}{2}$ and $F(u) = 1$ for $u \geqslant \frac{1}{2}$. Now $E(\tau)$ is obtained through integration by parts:

$$E_{P^t}(\tau) = \int_0^{1/2} F'(u)\, u\, \mathbf{d}u$$
$$= (F(u) - 1)\, u \,|_0^{1/2} - \int_0^{1/2} (F(u) - 1)\, \mathbf{d}u$$
$$= -\int_0^{1/2} (F(u) - 1)\, \mathbf{d}u$$
$$= \int_0^{1/2} \left(1 - \frac{2u}{d}\right)^{t-1} (1 - 2u)\, \mathbf{d}u$$

$$= -\frac{d}{2t}\left(1-\frac{2u}{d}\right)^t (1-2u)\Big|_0^{1/2} - \int_0^{1/2}\left(-\frac{d}{2t}\left(1-\frac{2u}{d}\right)^t\right)(-2)\,\mathbf{d}u$$

$$=\frac{d}{2t}-\frac{d}{t}\int_0^{1/2}\left(1-\frac{2u}{d}\right)^t\,\mathbf{d}u$$

$$=\frac{d}{2t}-\frac{d}{t}\left[-\frac{d}{2(t+1)}\left(1-\frac{2u}{d}\right)^{t+1}\right]\Bigg|_0^{1/2}$$

$$=\frac{d}{2t}+\frac{d^2}{2t(t+1)}\left(\left(1-\frac{1}{d}\right)^{t+1}-1\right)$$

This proves (3) and completes the proof of the theorem. ▌

Note that as t grows the above lower bound approaches $d/2t$. Since the lower bound holds for one distribution that is not varying with t we can also get lower bounds on the expected total number of mistakes in the first t trials.

COROLLARY 3.1. *For each $d \geqslant 1$, let X_d, G_d and P be defined as above. Then for any prediction strategy Q, for each $d \geqslant 1$ there exists a target function $f_d \in G_d$ such that for all $t \geqslant 1$, the expected total number of mistakes in prediction on the first t random examples of f_d is at least*

$$\frac{d}{2}\left(\ln\frac{t}{d}-1\right).$$

Proof. Let U be the uniform distribution on G_d as above. By linearity of the expectation, when f is drawn at random from G_d according to U and the points $x_1, ..., x_t$ are drawn independently at random according to P, the expected total number of mistakes of Q on sam$((x_1, ..., x_t), f)$ is

$$\sum_{i=1}^{t} E_{P^i \times U}(\mathbf{M}_Q^i), \qquad (*)$$

the sum of the probabilities of a mistake for the individual trials. Therefore there exists a target function $f_d \in G_d$ such that when the points $x_1, ..., x_t$ are drawn independently at random according to P, the expected total number of mistakes of Q on sam$((x_1, ..., x_t), f)$ is at least (*). Hence it suffices to show that

$$(*) \geqslant \frac{d}{2}\left(\ln\frac{t}{d}-1\right).$$

Using the bound from Theorem 3.2,

$$\begin{aligned}(*) &\geqslant \frac{1}{2}\sum_{i=1}^{t}\left(\frac{d}{i}+\frac{d^2}{i(i+1)}\left(\left(1-\frac{1}{d}\right)^{i+1}-1\right)\right)\\ &=\frac{1}{2}\sum_{i=1}^{t}\left(\frac{d}{i}+\left(\frac{d^2}{i}-\frac{d^2}{i+1}\right)\left(\left(1-\frac{1}{d}\right)^{i+1}-1\right)\right)\\ &=\frac{1}{2}\left(\sum_{i=1}^{t}\frac{d}{i}+\sum_{i=1}^{t}\frac{d^2}{i}\left(1-\frac{1}{d}\right)^{i+1}-\sum_{i=1}^{t}\frac{d^2}{i+1}\left(1-\frac{1}{d}\right)^{i+1}\right.\\ &\qquad\left.-\sum_{i=1}^{t}\frac{d^2}{i}+\sum_{i=1}^{t}\frac{d^2}{i+1}\right).\end{aligned}$$

We approximate the first sum and rewrite the second and third sum. The last two sums cancel except for two summands:

$$\begin{aligned}(*) &\geqslant \frac{1}{2}\left(d\ln(t)+d^2\left(1-\frac{1}{d}\right)\sum_{i=1}^{t}\left(1-\frac{1}{d}\right)^{i}\Big/ i\right.\\ &\qquad\left.-d^2\sum_{i=2}^{t+1}\left(1-\frac{1}{d}\right)^{i}\Big/ i-d^2+\frac{d^2}{t+1}\right).\end{aligned}$$

We now combine similar terms of the remaining sums:

$$\begin{aligned}(*) &\geqslant \frac{1}{2}\left(d\ln(t)+d^2\left(1-\frac{1}{d}\right)^2+\left(d^2\left(1-\frac{1}{d}\right)-d^2\right)\sum_{i=2}^{t}\left(1-\frac{1}{d}\right)^{i}\Big/ i\right.\\ &\qquad\left.-d^2\left(1-\frac{1}{d}\right)^{t+1}\Big/(t+1)-d^2+d^2/(t+1)\right).\end{aligned}$$

We drop the last and third to the last summands and simplify:

$$\begin{aligned}(*) &\geqslant \frac{1}{2}\left(d\ln(t)+(d-1)^2-d\sum_{i=2}^{t}\left(1-\frac{1}{d}\right)^{i}\Big/ i-d^2\right)\\ &=\frac{1}{2}\left(d\ln(t)+d^2-2d+1-d\sum_{i=1}^{t}\left(1-\frac{1}{d}\right)^{i}\Big/ i+d-1-d^2\right)\\ &=\frac{1}{2}\left(d(\ln(t)-1)-d\sum_{i=1}^{t}\left(1-\frac{1}{d}\right)^{i}\Big/ i\right).\end{aligned}$$

We use the fact that ln(1-a) $= -\sum_{i=1}^{\infty} a^i/i$ (for $|a|<1$) with $a = 1-1/d$.

$$\begin{aligned}(*) &\geqslant \frac{1}{2}\left(d(\ln(t)-1)+d\ln\left(\frac{1}{d}\right)\right)\\ &=\frac{d}{2}\left(\ln\left(\frac{t}{d}\right)-1\right). \quad ∎\end{aligned}$$

The concept class used in the above lower bounds consists of disjoint unions of d initial segments. By arranging the initial segments in various ways in $\mathbf{R}^d$ and concentrating P on those segments we immediately get lower bounds for other geometric concept classes. For example, consider the concept class consisting of indicator functions for all negative halfspaces in $\mathbf{R}^d$. (Negative halfspaces are halfspaces whose intersection with the negative portion of the first axis is infinite, and whose intersection with the positive portion of that axis is empty or finite.) This class has VC dimension d (see [HW87]). (Since we are giving a lower bound, the argument also applies to the class of all halfspaces in $\mathbf{R}^d$; this class has VC dimension $d+1$.) Let S_i, for $1 \leqslant i \leqslant d$, be the closed interval between 1 and 2 on the ith axis in $\mathbf{R}^d$. Let P be the distribution concentrated uniformly on $\bigcup_{i=1}^{d} S_i$. Now we simply embed the ith initial segment into S_i. If we let $h_{(q_1, \ldots, q_d)}$ denote the function from $\mathbf{R}^d$ to $\{0, 1\}$ defined by $h_{(q_1, \ldots, q_d)}(x_1, \ldots, x_d) = 1$ if and only if $(x_1, \ldots, x_d) \cdot (1/(q_1+1), \ldots, 1/(q_d+1)) \leqslant 1$, then for $(q_1, \ldots, q_d) \in [0, 1)^d$ the function $h_{(q_1, \ldots, q_d)}$ takes the value 1 on a halfspace that cuts an initial segment from each S_i. Let $H_d = \{h_{\bar{q}} : \bar{q} \in [0, 1)^d\}$ be the family of indicator functions of negative halfspaces defined in this way. Theorem 3.2 and Corollary 3.1 clearly hold if P is as described above, G_d is replaced by H_d, and U is the uniform distribution on H_d. This gives lower bounds for predicting negative halfspaces.

Similar constructions give lower bounds for the class of indicator functions for orthogonal rectangles in $\mathbf{R}^d$ (we embed two initial segments growing away from each other in each dimension) and the class of indicator functions for unions of d intervals in $\mathbf{R}$ (we embed d pairs of initial segments in the real line, where the segments of each pair grow away from each other). These classes each have VC dimension $2d$. Thus Theorem 3.2 and Corollary 3.1 hold for these concept classes as well, with d replaced by $2d$. In each of these cases, the algorithm-independent lower bounds of Theorem 3.2 and Corollary 3.1 are asymptotically within a factor of $\frac{1}{2}$ of the upper bounds on the prediction performance of the 1 inclusion graph strategy as given in Theorem 2.3 and Corollary 2.2, and of the computationally efficient prediction strategies for these classes given in examples 2.1, 2.3 and 2.4.

4. Prediction Strategies Using Hypothesis Spaces of Small VC Dimension

The results of the previous sections indicate that one approach to constructing good prediction strategies is to construct strategies with small permutation mistake bounds. Here we consider another approach, based on the learning results of Vapnik [Vap82] and Blumer *et al.* [BEHW89], in which a prediction is made on the basis of a consistent hypothesis chosen from a hypothesis space of small Vapnik–Chervonenkis dimension.

DEFINITION. For any prediction strategy Q and sample $s \in S_F$, the *hypothesis of Q generated by s*, is the function $h: X \rightarrow \{0, 1\}$ defined by $h(x) = Q(s, x)$. A hypothesis h is said to be *consistent* if it correctly labels the points of the sample s generating it, i.e., if for all pairs $(x, a) \in s$, $h(x) = a$.

THEOREM 4.1. *Let H be some collection of functions from X to $\{0, 1\}$ with* $\mathrm{VCdim}(H) = d \geqslant 1$. *Suppose that for all samples in S_F the hypothesis of prediction strategy Q is an element of H. Also assume that Q always chooses a hypothesis consistent with the sample. Then* $\hat{\mathbf{M}}_{Q,F}(t) \leqslant (2(d+1)/t) \log_2(4et/d)$ *for all $t > d$.*

Unlike in Theorem 2.3, in this bound there is no attempt to minimize the constants. We focus instead on providing a short proof. Before establishing this result, we prove the following lemma.

DEFINITION. Let Q be a prediction strategy for F. For each $t \geqslant 1, f \in F$, distribution P on X and $\bar{x} \in X^t$, $\mathbf{ER}^t_{Q,f,P}(\bar{x}) = P\{x \in X: Q(\mathrm{sam}(\bar{x}, f), x) \neq f(x)\}$.

Thus $\mathbf{ER}^t_{Q,f,P}(\bar{x})$ is the probability that the hypothesis of Q generated by $\mathrm{sam}(\bar{x}, f)$ disagrees with the target function f on a randomly drawn point. This is the notion of the "error" of a hypothesis used in Valiant's learning model [HKLW91, Val84].

LEMMA 4.1. $E_{P^t}(\mathbf{ER}^t_{Q,f,P}) = E_{P^{t+1}}(\mathbf{M}^{t+1}_{Q,f})$ *for any prediction strategy Q, target function $f \in F$, distribution P on X and $t \geqslant 0$.*

Proof.

$$E_{P^{t+1}}(\mathbf{M}^{t+1}_{Q,f}) = \int_{X^{t+1}} \mathbf{M}^{t+1}_{Q,f}(x_1, ..., x_{t+1})\, \mathbf{d}P^{t+1}(x_1, ..., x_{t+1})$$

$$= \int_{X^t} \left(\int_X \mathbf{M}^{t+1}_{Q,f}(x_1, ..., x_{t+1})\, \mathbf{d}P(x_{t+1}) \right) \mathbf{d}P^t(x_1, ..., x_t)$$

by Fubini's theorem. However, for fixed $(x_1, ..., x_t)$,

$$\int_X \mathbf{M}^{t+1}_{Q,f}(x_1, ..., x_{t+1})\, \mathbf{d}P(x_{t+1}) = \mathbf{ER}^t_{Q,f,P}(x_1, ..., x_t)$$

by definition. Hence

$$E_{P^{t+1}}(\mathbf{M}_{Q,f}^{t+1}) = \int_{X^t} \mathbf{ER}_{Q,f,P}^t(x_1, ..., x_t)\, \mathbf{d}P^t(x_1, ..., x_t)$$
$$= E_{P^t}(\mathbf{ER}_{Q,f,P}^t). \quad \blacksquare$$

Proof of Theorem 4.1. Let $d = \mathrm{VCdim}(H)$. Since Q is consistent and uses hypotheses in H, it follows from Theorem A2.1 and Proposition A2.1 of [BEHW89] (see [VC71, Vap82]) that for all $f \in F$, distributions P on X, $\varepsilon > 0$ and $t > d$,

$$P^t\{\bar{x} : \mathbf{ER}_{Q,f,P}^t(\bar{x}) \geqslant \varepsilon\} \leqslant 2(2et/d)^d\, 2^{-\varepsilon t/2}.$$

Since $\mathbf{ER}_{Q,f,P}^t \leqslant 1$, this implies that $E_{P^t}(\mathbf{ER}_{Q,f,P}^t) \leqslant \varepsilon + 2(2et/d)^d\, 2^{-\varepsilon t/2}$ for all $\varepsilon > 0$. Letting $\varepsilon = (2/t)(\log_2(t/d) + d \log_2(2et/d))$, it follows that $E_{P^t}(\mathbf{ER}_{Q,f,P}^t) \leqslant (2/t)(\log_2(t/d) + d \log_2(2et/d) + d) \leqslant (2(d+1)/t) \log_2(4et/d)$. The result then follows from Lemma 4.1. $\blacksquare$

Note that to be always able to find consistent hypotheses in H, VCdim(H) must be at least VCdim(F). Thus for sufficiently large t, the bound of Theorem 4.1 is greater, by at least a factor proportional to log(t/VCdim(F)), than the bound for an algorithm with an optimal permutation mistake bound. The following theorem shows that this gap is real by demonstrating that the bound in Theorem 4.1 is tight to within a constant factor.

THEOREM 4.2. *There exists a family of countable domains* $\{X_d\}_{d=1}^{\infty}$ *and corresponding concept classes* $\{F_d\}_{d=1}^{\infty}$ *with* $\mathrm{VCdim}(F_d) = d$ *for which there is a prediction strategy LEGAL such that*

(1) *when LEGAL is presented with a sample from* S_{F_d} *it chooses a consistent hypothesis from* F_d, *and*

(2) *for any* $d \geqslant 1$

$$\hat{\mathbf{M}}_{LEGAL, F_d}(t) \geqslant (1 - 1/e - o(1)) \frac{d}{t} \ln \frac{t}{d}.$$

Here $o(1)$ represents a function of t that goes to zero as $t \to \infty$.

Proof. We first describe the function class and the prediction strategy *LEGAL* for which the theorem holds. Let the domain X_d consist of d disjoint copies of the natural numbers, which we will call *types*. The function class F_d consists of all functions that are 1 on at most one point in each type. It is easy to see that the VC dimension of F_d is d.

Given sample $s = ((x_1, a_1), ..., (x_{t-1}, a_{t-1}))$ and an unlabeled point x_t, *LEGAL* predicts as follows: if any point in x_t's type occurs in s with label

1, then *LEGAL* predicts 1 iff x_t is this point, else *LEGAL* predicts 1 iff x_t is the smallest number in its type that does not occur in s. Note that for any sample s in S_{F_d}, the hypothesis of *LEGAL* is 1 on exactly one point from each type and is thus in F_d. Clearly the hypothesis is also consistent with the sample s.

Recall that $\hat{\mathbf{M}}_{LEGAL,F_d}(t) = \sup E_{P^t}(\mathbf{M}^t_{LEGAL,f})$ over all $f \in F_d$ and distributions P on X_d. For each t, to obtain a lower bound $b(t)$ for $\hat{\mathbf{M}}_{LEGAL,F_d}(t)$, it suffices to exhibit a function $f \in F_d$ and a distribution $P_{t,d}$ on X_d such that $E_{P^t_{t,d}}(\mathbf{M}^t_{LEGAL,f}) \geqslant b(t)$. To obtain our bound, for all t, with $t+1 \geqslant d$, we will let f be the constant function 0 and $P_{t+1,d}$ be the distribution that is uniform over the union of the sets $\{1, ..., n\}$ from all d types and 0 elsewhere, where

$$n = \left\lceil \frac{t}{d(\ln(t/d) - \ln\ln(t/d))} \right\rceil.$$

For each t, let $p(t)$ be the probability that *LEGAL* makes a mistake on the tth trial, i.e., $p(t) = E_{P^t_{t,d}}(\mathbf{M}^t_{LEGAL,f})$. We will obtain a lower bound for $p(t+1)$. For the target function f, *LEGAL* makes a mistake on the point x_{t+1} only when x_{t+1} is the smallest number of its type that does not occur in the previous t points $\{x_1, ..., x_t\}$. All types are equally likely, so we may condition this probability on x_{t+1} being any particular type, say type 1. Thus $p(t+1)$ is the probability that x_{t+1} is the smallest number of type 1 that does not occur in $\{x_1, ..., x_t\}$, given that x_{t+1} is type 1. Clearly $p(t+1) = q(t)/n$, where $q(t)$ is the probability that there is at least one of the first n numbers of type 1 that does not occur in $\{x_1, ..., x_t\}$.

For each t let us define the random variable $\mathbf{X}^t$, where $\mathbf{X}^t(x_1, ..., x_t)$ is the number of type 1 numbers from $\{1, ..., n\}$ that do not occur in $\{x_1, ..., x_t\}$. Thus $q(t) = 1 - \text{Prob}(\mathbf{X}^t = 0)$. We will show that for any d, as $t \to \infty$, the distribution of $\mathbf{X}^t$ approaches the Poisson distribution with parameter $\lambda = 1$ (see, e.g., [Fel68]). Hence $\text{Prob}(\mathbf{X}^t = 0)$ approaches $1/e$. Since $1/n = (d/t)(\ln(t/d) - o(\ln(t/d))$, this gives our result.

It remains to show $\mathbf{X}^t$ converges pointwise to the Poisson distribution. For each i, $1 \leqslant i \leqslant n$, let $\mathbf{X}^t_i$ be the random variable that is 1 if the number i from the type 1 numbers does not occur in $\{x_1, ..., x_t\}$, else 0. Thus $\mathbf{X}^t = \sum_{i=1}^{n} \mathbf{X}^t_i$. Now for each r, $1 \leqslant r \leqslant n$, let S^t_r be the rth binomial moment of $\mathbf{X}^t$; i.e., S^t_r is the sum over all subsets of $\{1, ..., n\}$ of size r of the probability that no number in that subset occurs in $\{x_1, ..., x_t\}$. It is easily verified using the standard inclusion/exclusion arguments (the Bonferroni inequalities) that if

$$\lim_{t \to \infty} S^t_r = \frac{\lambda^r}{r!}$$

for all r then $\mathbf{X}^t$ converges pointwise to the Poisson distribution with parameter λ, i.e., $\lim_{t\to\infty} \text{Prob}(\mathbf{X}^t = k) = e^{-\lambda}\lambda^k/k!$ for all $k \geqslant 0$ (see, e.g., [P85, Appendix V]).

Let A be any set composed of r of the first n type 1 numbers. Since each point in A has probability $1/nd$, the probability that none of the points $\{x_1, ..., x_t\}$ are in A is $(1 - r/nd)^t$. Hence

$$S_r^t = \binom{n}{r}\left(1 - \frac{r}{nd}\right)^t.$$

It suffices to show that $\lim_{t\to\infty} r!\, S_r^t = 1$. Taking logs and simplifying, this reduces to

$$\left(\sum_{i=0}^{r-1} \ln(n-i)\right) + t \ln\left(1 - \frac{r}{nd}\right) \to 0.$$

Since r is fixed and n goes to infinity, $(\sum_{i=0}^{r-1} \ln(n-i)) - r \ln n \to 0$. Thus it suffices to show that

$$r \ln n + t \ln\left(1 - \frac{r}{nd}\right) \to 0.$$

Let

$$n' = \frac{t}{d(\ln(t/d) - \ln \ln(t/d))}.$$

For sufficiently large t,

$$t \ln(1 - r/n'd) + tr/n'd \leqslant t \ln(1 - r/nd) + tr/nd \leqslant 0.$$

It can also be shown that $t \ln(1 - r/n'd) + tr/n'd \to 0$ as $t \to \infty$ by l'Hôpital's rule. Hence $tr/nd + t \ln(1 - r/nd) \to 0$. Thus it suffices to show that

$$r \ln n - \frac{tr}{nd} \to 0,$$

or equivalently that

$$\ln n - \frac{t}{nd} \to 0.$$

Let $g(t) = \ln(t/d) - \ln\ln(t/d)$, and write $n = t/\mathrm{d}g(t) + \gamma(t)$, where $0 \leqslant \gamma(t) \leqslant 1$. We first show that

$$\frac{t}{nd} - g(t) \to 0.$$

This can be rewritten as

$$\frac{1}{1/g(t)+d\gamma(t)/t}-g(t)\to 0.$$

This is equivalent to

$$\frac{-\gamma(t)}{t/dg^2(t)+\gamma(t)/g(t)}\to 0,$$

which holds since $g(t)$ is logarithmic in t. It remains to show that $\ln n - g(t)\to 0$, which is easily verified. ∎

In this paper we have discussed two methods for obtaining expected mistake bounds for a prediction algorithm. The first method bounds the probability of a mistake in terms of the permutation mistake bound (Corollary 2.1). The second method predicts with a consistent hypothesis and bounds the probability of a mistake in terms of the VC dimension of the hypothesis class (Theorem 4.1). Neither of the two methods subsumes the other. In particular, there exists a target class F for which the following are true:

(i) There exists an algorithm Q for predicting F that uses a hypothesis space with infinite Vapnik–Chervonenkis dimension for which $\hat{\mathbf{M}}_{Q,F}(t)=1/t$.

(ii) There exists an algorithm Q' for predicting F that chooses consistent hypotheses from a space with Vapnik–Chervonenkis dimension 1 for which $\hat{\mathbf{M}}_{Q',F}(t)$ is 1.

To see this, take the domain X to be the positive integers and let the concept class F consist of all functions from X to $\{0,1\}$ that are 1 on at most one point. The Vapnik–Chervonenkis dimension of F is 1. Consider the following two prediction algorithms for F. In each case we assume that the algorithm has received as input a sample $(x_1,...,x_{t-1})\in X^{t-1}$ labeled according to some concept $f\in F$. The algorithm has also received another point $x_t\in X$ whose label it is to predict.

The algorithm Q always predicts 0 except in the following two cases:

(1) It predicts 1 when it must do so in order to be consistent with previous examples.

(2) It predicts 1 when all of $x_1,...,x_{t-1}$ are odd numbers, $f(x_i)=0$ for all $i\in\{1,...,t-1\}$, and $x_t=2x_i$ for some $i\in\{1,...,t-1\}$.

The algorithm Q' is the algorithm LEGAL described above, in the case $d = 1$. This algorithm also predicts 0 except in two cases:

(1) This case is the same as case (1) for the previous algorithm.

(2) It predicts 1 when x_t is the least positive integer not contained in $\{x_1, ..., x_{t-1}\}$ and $f(x_i) = 0$ for all $i \in \{1, ..., t-1\}$.

The algorithm Q has a permutation mistake bound of $1/t$ but uses a hypothesis space that includes all finite subsets of the positive integers of the form $4n + 2$ for some $n \geqslant 0$, and hence has infinite VC dimension. The algorithm Q' chooses consistent hypotheses from a space with Vapnik–Chervonenkis dimension 1 but its permutation mistake bound is 1 (yielding no interesting mistake bound). (To see the latter, consider the case where $\{x_1, ..., x_t\} = \{1, ..., t\}$ and f is 0 on all of these points.) Thus Corollary 2.1 gives a non-trivial upper bound only for algorithm Q, while the upper bound of Theorem 4.1 is finite only for algorithm Q'.

5. PAC Learning Algorithm Derived from the 1-Inclusion Graph Strategy

In this section we show how the 1-inclusion graph prediction strategy of Section 2 leads to a PAC learning algorithm requiring $O((\text{VCdim}(F)/\varepsilon) \log(1/\delta))$ examples. Since the introduction of PAC learning algorithms by Valiant [Val84] a number of different definitions have been used. We choose a definition that is convenient for our purposes and refer to [HKLW91] for a discussion of equivalent definitions.

Let F be a concept class over some domain X. Throughout this section we assume $\text{VCdim}(F) \geqslant 1$. If $f \in F$ is a given target concept, h is a $\{0, 1\}$-valued function over X, and P any fixed probability distribution on X, then define the *error*[8] of h (with respect to f and P) as the probability that f disagrees with h on an example $(x, f(x))$, where x is drawn from X according to P. Intuitively, a learning algorithm is to output a hypothesis that has small error with high probability by drawing a polynomially sized sample of the target function independently at random according an arbitrary but fixed distribution. This is formalized by having two parameters $0 < \varepsilon, \delta < 1$ that are given to the learning algorithm, along with access to a source of random examples of the target function, and by requiring the following *PAC-criterion*: the hypothesis output by the

[8] This is consistent with our previous usage of error. Recall that in Section 4 we defined $\mathbf{ER}'_{Q,f,P}(\bar{x})$ to be the error of the hypothesis of the prediction strategy Q after the sample $\text{sam}(\bar{x}, f)$ is received.

algorithm must have error at most ε with probability at least $1-\delta$. The latter probability is over the random choice of the examples, and any randomization inherent in the algorithm.

DEFINITION. A *PAC-learning algorithm* A for F is an algorithm that for all $0<\varepsilon$, $\delta<1$, for all $f\in F$, and for all probability distributions P on X, outputs a representation of a hypothesis fulfilling the PAC-criterion by drawing a sample of f of size at most $t(1/\varepsilon, 1/\delta)$ independently at random according to P, where t is a polynomial.

Call $t(1/\varepsilon, 1/\delta)$ the *sample complexity* of A. Note that usually it is also required that the hypothesis be of a particular form and that the PAC-learning algorithm run in polynomial time. We will ignore these issues in this section.

We now define a PAC Algorithm A that learns any concept class F with sample complexity $O((\mathrm{VCdim}(F)/\varepsilon)\log(1/\delta))$. The algorithm A first runs the deterministic 1-inclusion graph strategy of Section 2 exactly $\lceil\log_2(2/\delta)\rceil$ times, each time using a new sample of size $\lceil 4\mathrm{VCdim}(F)/\varepsilon\rceil$. After processing each sample the final hypothesis is put in the set G. In the second step of the algorithm we run the following procedure with a new sample of size $\lceil(32/\varepsilon)(\ln(2/\delta)+\ln(\lceil\log_2(2/\delta)\rceil+1))\rceil$:

PROCEDURE $MI(G, s)$.

Parameters: a finite set G of hypotheses and a sample $s=(x_1, a_1)$, (x_2, a_2), ..., (x_t, a_t).

Output: the hypothesis $h\in G$ that has the least number of inconsistencies with the sample, that is, such that $|\{j: a_j\neq h(x_j)\}|$ is minimum. (If there is a tie, any scheme can be used to break it.)

(This procedure is a standard hypothesis-testing procedure. The description and analysis here are taken from [L89b].)

Clearly the sample complexity of this algorithm is $O((\mathrm{VCdim}(F)/\varepsilon)$ $\log(1/\delta))$. We still have to show that the hypothesis it outputs fulfills the PAC-criterion. First observe that by Theorem 2.2 and Lemma 4.1 the expected error of each hypothesis of G is at most $\varepsilon/4$. Thus by Markov's Lemma each hypothesis of G has error larger than $\varepsilon/2$ with probability less than $1/2$. Since a different sample is used for each hypothesis of G, this holds independently for each of the $\lceil\log_2(2/\delta)\rceil$ hypotheses in G. It follows that G has a hypothesis of error at most $\varepsilon/2$ with probability larger than $1-\delta/2$. By the following lemma of [L89b], if G has a hypothesis of error at most $\varepsilon/2$ then MI, when executed with a sample of size $\lceil (32/\varepsilon)(\ln(2/\delta)+\ln(\lceil \log_2(2/\delta)\rceil+1))\rceil$, outputs with probability at least $1-\delta/2$ a hypotheses of G with error at most ε.

LEMMA 5.1. *Fix any target function f and a probability distribution P on the domain of f with respect to which the error of the hypotheses is defined. Suppose $0<\varepsilon<1$ and let G be a finite set of hypotheses containing at least one hypothesis with error at most $\varepsilon/2$. Then if $x_1, x_2, ..., x_t$ are drawn independently at random according to P, with probability at least $1-(|G|+1)\,e^{-\varepsilon t/32}$, the error of the hypothesis output by MI is at most ε when given the sample $s=(x_1, f(x_1)), (x_2, f(x_2)), ..., (x_t, f(x_t))$.*

We conclude that the hypothesis output by the above algorithm fulfills the PAC-criterion. We thus obtain the following theorem:

THEOREM 5.1. *For any concept class F of VC dimension at least one, there is a PAC-learning algorithm with sample complexity $O((\text{VCdim}(F)/\varepsilon)\log(1/\delta))$.*

6. CONCLUSION AND OPEN PROBLEMS

We have introduced a computational model of learning related to the PAC model, but focusing on minimizing the probability of mistakes in prediction instead of producing a hypothesis in a specific form. We have characterized the best achievable performance bounds for prediction strategies in this model, at least to within a small constant factor, and developed prediction strategies that achieve them. Finally, we have compared the performance of these new prediction strategies to those obtained by the "standard" PAC strategy of learning by finding consistent (or "almost" consistent) hypotheses in a hypothesis space of small VC dimension. This comparison has shown that our method of using the 1-inclusion graph to directly minimize the probability of predictive error achieves improved predictive performance. The prediction strategy that is based on the 1-inclusion graph also leads to a PAC learning algorithm that for some ranges of the parameters requires a smaller sample size than previous algorithms. Furthermore, the design techniques and analytic tools we develop here, especially those involving the permutation mistake bound, may also be useful in designing and analyzing other learning algorithms in situations where efficient on-line predictive performance is an issue (e.g., in robotics).

However, before serious consideration of practical applications, there is still much to be done to make our model more flexible and more general. One step would be to extend our measure of predictive performance, and our results, to the more general measures used in statistical decision theory, in which the prediction algorithm can choose from a variety of responses, and receives a real-valued reward or penalty for each response depending on the current situation. An extension of the PAC model along these lines is suggested in [H90,92].

Even more important will be to generalize the prediction model to allow various types of noise in the labels of the examples. Here there are also many approaches, from the non-probabilistic notion of "anomalies" studied in [L89a] to the random noise model of [AL87] or the more general stochastic models of [KS94, H90,92]. These latter noise models are taken from work in pattern recognition and statistical inference, in which random examples are generated directly from a distribution on the set of all possible (labeled) examples. It is especially important to generalize the results on prediction strategies to handle this case insofar as it is possible.

Another important direction for further research involves weakening the assumption that the target function and domain distribution are chosen by an adversary. It would be nice to have a good general, distribution specific analysis of the probability of a mistake at trial t, even if it was still worst case over all possible target concepts in a given concept class F. However, we could go still further and assume that the target concepts in F are selected at random according to some specific distribution as well. This would lead to a Bayesian analysis, instead of the minimax analysis used here. Some results in this direction are given in [HKS94].

A number of smaller technical issues remain as well. Of these, the most intriguing is the question of the best possible constant for Theorem 2.3. Can we show that for any concept class F there exists a randomized prediction strategy Q with $\hat{\mathbf{M}}_{Q,F}(t) \leq c_0 \mathrm{VCdim}(F)/t$, where $c_0 < 1$? Theorem 3.2 shows that this is not possible with $c_0 < \frac{1}{2}$, but this still leaves some gap.

However, as with the PAC learning model, the most significant open problems that remain concern the existence of computationally efficient learning/prediction algorithms. The time complexity of the 1-inclusion graph strategy grows exponentially as $\mathrm{VCdim}(F)$ increases. Thus this strategy is not computationally feasible for many target classes for which the VC dimension grows polynomially in the relevant parameters. Still, in some cases, such as intersection closed concept classes, unions of intervals, and half-spaces, our methods lead to essentially optimal prediction strategies with time complexities polynomial in the VC dimension.

ACKNOWLEDGMENTS

We thank Lenny Pitt, Michael Kearns, Bob Sloan, and Masahiro Matsuoka for helpful comments on this material. Also, we thank David Helmbold for helping us to simplify the upper bounds of Theorem 2.3 for the deterministic 1-inclusion graph strategy.

RECEIVED December 7, 1990; FINAL MANUSCRIPT RECEIVED July 17, 1992

REFERENCES

[AHW87] ALON, N., HAUSSLER, D., AND WELZL, E. (1987), Partitioning and geometric embedding of range spaces of finite Vapnik–Chervonenkis dimension, *in* "Proceedings, 3rd Symposium on Computational Geometry, Waterloo, June 1987," pp. 331–340.

[AT92] ALON, N., AND TARSI, M. (1992), Colorings and orientations of graphs, *Combinatorica* **12**, 125–134.

[A87] ANGLUIN, D. (1987), Queries and concept learning, *Mach. Learning* **2**, No. 4, 319–342.

[A90] ANGLUIN, D. (1990), Negative results for equivalence queries, *Mach. Learning* **5**, 121–150.

[AL87] ANGLUIN, D., AND LAIRD, P. D. (1987), Identifying k-CNF formulas from noisy examples, *Mach. Learning* **2**, No. 4, 343–370.

[B88] BOUCHERON, S. (1988), Learnability from positive examples in the Valiant framework, unpublished manuscript.

[B90] BLUM, A. (1990), Separating distribution-free and mistake-bound learning models over the Boolean domain, *in* "Proceedings of the 31st Annual Symposium on Foundations of Computer Science," IEEE, pp. 211–218.

[BEHW87] BLUMER, A., EHRENFEUCHT, A., HAUSSLER, D., AND WARMUTH, M. K. (1987), Occam's razor, *Inform. Process. Lett.* **24**, 377–380.

[BEHW89] BLUMER, A., EHRENFEUCHT, A., HAUSSLER, D., AND WARMUTH, M. K. (1989), Learnability and the Vapnik–Chervonenkis dimension, *J. Assoc. Comput. Mach.* **36**, No. 4, 929–965.

[B72] BONDY, J. A. (1972), Induced subsets, *J. Combin. Theory Ser. B* **12**, 201–202.

[CF88] CHAZELLE, B., AND FRIEDMAN, J. (1988), A deterministic view of random sampling and its use in geometry, *Combinatorica* **10**, 229–249.

[CW89] CHAZELLE, B., AND WELZL, E. (1989), Quasi-optimal range searching and VC-dimensions, *Discrete Comput. Geom.* **4**, 467–490.

[E87] EDELSBRUNNER, H. (1987), "Algorithms in Combinatorial Geometry," Springer-Verlag, Berlin/New York.

[EGS88] EDELSBRUNNER, H., GUIBAS, L., AND SHARIR, M. (1988), The complexity of many faces in arrangements of lines and of segments, *in* "Proceedings, 4th Annual ACM Symposium on Computational Geometry, Urbana, IL," pp. 44–55.

[EH89] EHRENFEUCHT, A., AND HAUSSLER, D. (1989), Learning decision trees from random examples, *Inform. and Comput.* **82**, 231–246.

[EHKV89] EHRENFEUCHT, A., HAUSSLER, D., KEARNS, M., AND VALIANT L. (1989), A general lower bound on the number of examples needed for learning, *Inform. and Comput.* **82**, 247–261.

[Fel68] FELLER, W. (1968), "An Introduction to Probability Theory and Its Applications," 3rd ed., Wiley, New York.

[F89] FLOYD, S. (1989), Space-bounded learning and the Vapnik–Chervonenkis dimension, *in* "Proceedings of the Second Annual Workshop on Computational Learning Theory, Santa Cruz, CA" (R. Rivest, D. Haussler, and M. K. Warmuth, Eds.), pp. 349–364, Morgan Kaufmann, San Mateo, CA.

[F95] FREUND, Y. (1995), Boosting a weak learning algorithm by majority, *Inform. and Comput.*, to appear.

[GRS93] GOLDMAN, S. A., RIVEST, R. L., AND SCHAPIRE, R. E. (1993), Learning binary relations and total orders, *SIAM J. Comput.* **22**, No. 5, 1006–1034.

[GKS95] GOLDMAN, S., KEARNS, M. J., AND SCHAPIRE, R. E. (1995), On the sample complexity of weak learning, *Inform. and Comput.*, to appear.

[H88a] HAUSSLER, D. (1988a), Quantifying inductive bias: AI learning algorithms and Valiant's learning framework, *Artif. Intell.* **38**, pp. 177–221.

[H88b] HAUSSLER, D. (1988b), "Space Efficient Learning Algorithms," UCSC Technical Report UCSC-CRL-88-06, Department of Computer and Information Sciences, University of California at Santa Cruz.

[H90] HAUSSLER, D. (1990), Decision theoretic generalizations of the PAC learning model, *in* "ALT '90," pp. 21–41.

[H95] HAUSSLER, D. (1995), Sphere Packing Numbers for Subsets of the Boolean n-cube with Bounded Vapnik–Chervonenkis Dimension, *J. Combin. Theory Ser. A*, to appear.

[H92] HAUSSLER, D. (1992), Decision theoretic generalizations of the PAC model for neural net and other learning applications, *Inform. and Comput.* **100**, 78–150.

[HKLW91] HAUSSLER, D., KEARNS, M., LITTLESTONE, N., AND WARMUTH, M. K. (1991), Equivalence of models for polynomial learnability, *Inform. and Comput.* **95**, 129–161.

[HKS94] HAUSSLER, D., KEARNS, M., AND SCHAPIRE, R. (1994), Bounds on the sample complexity of Bayesian learning using information theory and the VC dimension, *Mach. Learning* **14**, No. 1, 83–113.

[HLW90] HAUSSLER, D., LITTLESTONE, N., AND WARMUTH, M. K. (1990), "Predicting $\{0, 1\}$-functions on randomly drawn points," Technical Report UCSC-CRL-90-54, Department of Computer Science, University of California, Santa Cruz, CA.

[HW87] HAUSSLER, D., AND WELZL, E. (1987), Epsilon-nets and simplex range queries, *Discrete Comput. Geom.* **2**, 127–151.

[HSW90] HELMBOLD, D., SLOAN, R., AND WARMUTH, M. K. (1990), Learning nested differences of intersection-closed concept classes, *Mach. Learning* **5**, 165–196.

[HSW92] HELMBOLD, D., SLOAN, R., AND WARMUTH, M. K. (1992), Learning integer lattices, *SIAM J. Comput.* **21**, No. 2.

[K84] KARMARKAR, N. (1984), A new polynomial-time algorithm for linear programming, *Combinatorica* **4**, 373–395.

[KLPV87] KEARNS, M., LI, M., AND VALIANT, L. (1987), Learning Boolean formulae, *J. Assoc. Comput. Mach.*, to appear.

[KPG92] KOMLOS, J., PACH, J., AND WOEGINGER, G. (1992), Almost tight bounds for epsilon-nets, *Discrete Comput. Geom.* **7**, 163–173.

[KS94] KEARNS, M. J., AND SCHAPIRE, R. E. (1994), Efficient distribution-free learning of probabilistic concepts, *J. Comp. Systems Sci.* **48**, No. 3, 464–497.

[KV92] KEARNS, M., AND VALIANT, L. (1992), Cryptographic limitations on learning Boolean formulae and finite automata, *J. Assoc. Comput. Mach.* **41** No. 1, 67–95.

[L87] LITTLESTONE, N. (1987), Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm, *Mach. Learning* **2**, No. 4, 285–318.

[L89a] LITTLESTONE, N. (1989a), "Mistake Bounds and Logarithmic Linear-Threshold Learning Algorithms," Ph.D. thesis, University of California, Santa Cruz, CA.

[L89b] LITTLESTONE, N. (1989b), From on-line to batch learning, *in* "Proceedings of the Second Annual Workshop on Computational Learning Theory, Santa Cruz, CA" (R. Rivest, D. Haussler, and M. K. Warmuth, Eds.), pp. 269–284, Morgan Kaufmann, San Mateo, CA.

[LW94] LITTLESTONE, N., AND WARMUTH, M. (1994), The weighted majority algorithm, *Inform. and Comput.* **108**, 212–261.

[MSW90] MATOUSEK, J., SEIDEL, R., AND WELZL, E. (1990), How to net a lot with a little: Small epsilon-nets for disks and halfspaces, *in* "Proceedings, 6th ACM Symposium on Computational Geometry, Berkeley, CA," pp. 16–22.

[MT92] MAASS, W., AND TURAN, G. (1992), Lower bound methods and separation results for on-line learning models, *Mach. Learning* **9**, 107–145.

[MT94a] MAASS, W., AND TURAN, G. (1994), How fast can a threshold gate learn?, *in* "Computational Learning Theory and Natural Learning Systems" (Drastal, Hanson, and Rivest, Eds.), MIT Press, 381–414.

[MT94b] MAASS, W., AND TURAN, G. (1994b), Algorithms and lower bounds for on-line learning of geometrical concepts, *Mach. Learning* **14**, 251–269.

[N87] NATARAJAN, B. K. (1987), On learning boolean functions, *in* "Proceedings of the 19th Annual ACM Symposium on Theory of Computing, New York," pp. 296–304.

[OH91] OPPER, M., AND HAUSSLER, D. (1991), Calculation of the learning curve of Bayes optimal classification algorithm for learning a perceptron with noise, *in* "Proceedings of the Fourth Workshop on Computational Learning Theory, Santa Cruz," pp. 75–87, Morgan Kaufmann, San Mateo, CA.

[P84] POLLARD, D. (1984), "Convergence of Stochastic Processes," Springer-Verlag, New York.

[P85] PALMER, E. (1985), "Graphical Evolution," Wiley, New York.

[PW90] PITT, L., AND WARMUTH, M. K. (1990), Prediction-preserving reducibility, *J. Comp. System Sci.* **41**, 430–467.

[PV88] PITT. L., AND VALIANT, L. G. (1988), Computational limitations on learning from examples, *J. Assoc. Comput. Math.* **35**, 965–984.

[R87] RIVEST, R. L. (1987), Learning decision-lists, *Mach. Learning* **2**, No. 3, 229–246.

[Sau72] SAUER, N. (1972), On the density of families of sets, *J. Combin. Theory Series A* **13**, 145–147.

[Sch90] SCHAPIRE, R. E. (1990), The strength of weak learnability, *Mach. Learning* **5**, 165–196.

[Sei91] SEIDEL, R. (1991) Small-dimensional Linear programming and convex hulls made easy, *Discrete Comput. Geom.* **7**, 423–434.

[Shv88] SHVAYTSER, H. (1988), Linear manifolds are learnable from positive examples, unpublished manuscript.

[Vap82] VAPNIK, V. N. (1992), Estimation of Dependences Based on Empirical Data," Springer-Verlag, New York.

[Val84] VALIANT, L. G. (1984), A theory of the learnable, *Comm. ACM* **27**, No. 11, 1134–1142.

[Val85] VALIANT, L. G. (1985), Learning disjunctions of conjunctions, *in* "Proceedings, 9th IJCAI, Los Angeles, CA.," Vol. 1, pp. 560–566.

[VC71] Vapnik, V. N., and Chervonenkis, A. Ya. (1971), On the uniform convergence of relative frequencies of events to their probabilities, *Theory Probab. Appl.* **16**, No. 2, 264–280.

[W86] Wilf, H. S. (1986), "Algorithms and Complexity," Prentice–Hall, Englewood Cliffs, NJ.

[W86] Wilf, H. S. (1986), "Algorithms and Complexity," Prentice–Hall, Englewood Cliffs, NJ.

[W88] Welzl, E. (1988), Partition trees for triangle counting and other range search problems, *in* "Proceedings, 4th Annual ACM Symposium on Computational Geometry, Urbana, IL," pp. 23–33.

[WD81] Wenocur, R. S., and Dudley, R. M. (1981), Some special Vapnik–Chervonenkis classes, *Discrete Math.* **33**, 313–318.