where

$$\nu_i = \begin{cases} \mu_i/c & \text{if } i \leq n \text{ and } \mu_i \geq 0 \\ -\mu_{i-n}/c & \text{if } n < i \leq 2n \text{ and } \mu_{i-n} < 0 \\ 1 - \frac{1}{c}\sum_i |\mu_i| & \text{if } i = 2n+1 \\ 0 & \text{otherwise.} \end{cases}$$

Note that for each i, ν_i is nonnegative, and that $\sum_i \nu_i = 1$. Choose $g \in \text{LINEAR}(n, M, c), \vec{x} \in [0, M]^n$. Let $\vec{\mu}$ be the coefficient vector of g, $I^+ = \{i : \mu_i > 0\}$ and $I^- = \{i : \mu_i \leq 0\}$. We have

$$\begin{aligned} \psi(g)(\phi(\vec{x})) &= \left(\sum_{i \in I^+} \frac{\mu_i(x_i + M)}{2cM}\right) + \left(\sum_{i \in I^-} \frac{-\mu_i(-x_i + M)}{2cM}\right) + \frac{1}{2}\left(1 - \frac{1}{c}\sum_i |\mu_i|\right) \\ &= \left(\frac{1}{2cM}\sum_{i=1}^{n} \mu_i x_i\right) + \left(\sum_i \frac{|\mu_i|}{2c}\right) + \frac{1}{2}\left(1 - \frac{1}{c}\sum_i |\mu_i|\right) \\ &= \frac{g(\vec{x})}{2cM} + 1/2. \end{aligned}$$

Thus there is a $2cM$-reduction from $\text{LINEAR}(n, M, c)$ to $\text{WA}(2n+1, 1, 0)$. The theorem now follows immediately from Theorem 81. The first bound can be proved by giving an M-reduction from $\text{WA}(n, M, \kappa)$ to $\text{WA}(n, 1, \kappa)$ along the lines of the reduction given above. The details are omitted. □

$$\begin{aligned}\sum_{t=1}^{\infty}(\psi(f)(\phi(x_t)) - \frac{\rho_t - k}{\alpha})^p &= \sum_{t=1}^{\infty}(\frac{f(x_t) - k}{\alpha} - \frac{\rho_t - k}{\alpha})^p \\ &= 1/\alpha^p \sum_{t=1}^{\infty}(f(x_t) - \rho_t)^p \\ &= N/\alpha^p\end{aligned}$$

we have

$$\sum_{t=1}^{\infty}(\lambda_t - \frac{\rho_t - k}{\alpha})^p \leq \mathrm{L}_p(A, \mathcal{G}, N/\alpha^p).$$

Hence,

$$\begin{aligned}\sum_{t=1}^{\infty}((\alpha\lambda_t + k) - \rho_t)^p &= \alpha^p \sum_{t=1}^{\infty}(\lambda_t - \frac{\rho_t - k}{\alpha})^p \\ &\leq \alpha^p \mathrm{L}_p(A, \mathcal{G}, N/\alpha^p).\end{aligned}$$

The theorem follows from the fact that S was chosen arbitrarily. □

This theorem is applied in the following section.

C.1 Proof of Theorem 8

We will prove only the second bound. The first can be proved analogously.

Choose n, M, c appropriately. We present a $2cM$-reduction from LINEAR(n, M, c) to WA$(2n+1, 1, 0)$. The theorem then follows immediately from Theorem 81 and Theorem 7.

Define the instance transformation $\phi : [0, M]^n \to [0, 1]^{2n+1}$ by

$$\phi(\vec{x}) = \left(\frac{x_1 + M}{2M}, ..., \frac{x_n + M}{2M}, \frac{-x_1 + M}{2M}, ..., \frac{-x_n + M}{2M}, \frac{1}{2}\right)$$

and define $\psi : \text{LINEAR}(n, M, c) \to \text{WA}(2n+1, 1, 0)$ as follows. If $g \in \text{LINEAR}(n, M, c)$ is defined by

$$g(\vec{x}) = \sum_{i=1}^{n} \mu_i x_i,$$

then let $\psi(g) = f$, where f is defined by

$$f(\vec{x}) = \sum_{i=1}^{2n+1} \nu_i x_i,$$

Appendix C. Reductions between real-valued learning problems

In this section, we describe a notion of reductions between real-valued learning problems. These transformations generalize the prediction preserving reductions that have been used in a similar manner in the learning of $\{0,1\}$-valued functions [Haussler, 1989b] [Littlestone, 1988] [Kearns *et al.*, 1987] [Pitt and Warmuth, 1990].

We will need the following definition. Let X and Y be sets, and let $\mathcal{F}$ and $\mathcal{G}$ be families of real-valued functions defined on X and Y respectively. Let $\alpha \geq 0$. We say that $\mathcal{F}$ α-reduces to $\mathcal{G}$ if and only if there is a function $\phi : X \to Y$, called an *instance transformation*, a function $\psi : \mathcal{F} \to \mathcal{G}$, called a *target transformation*, and $k \in \mathbf{R}$ such that for all $x \in X, f \in \mathcal{F}$,

$$f(x) = \alpha\psi(f)(\phi(x)) + k.$$

We are now ready for the following theorem, which gives loss bounds for a class of functions in terms of those for a class to which the function can be α-reduced.

Theorem 81: *Let X and Y be sets, and let $\mathcal{F}$ and $\mathcal{G}$ be families of real-valued functions defined on X and Y respectively. Let A be an algorithm for Y. Choose $p, \alpha, N \geq 0$. Then if $\mathcal{F}$ α-reduces to $\mathcal{G}$, there exists an algorithm B for X, such that*

$$L_p(B, \mathcal{F}, N) \leq \alpha^p L_p(A, \mathcal{G}, N/\alpha^p).$$

Proof: Define B as follows. Given an instance x, B feeds $\phi(x)$ to A, and if A predicts λ, B returns $\alpha\lambda + k$. Then, when B gets ρ as a reinforcement, it feeds $(\rho - k)/\alpha$ to A.

Choose $f \in \mathcal{F}$, and let $S = \langle (x_t, \rho_t) \rangle_{t \in \mathbf{N}}$ be a sequence of example-reinforcement pairs. Let $\langle \lambda_t \rangle_{t \in \mathbf{N}}$ be the sequence of predictions made by A on $\langle (\phi(x_t), (\rho_t - k)/\alpha) \rangle_{t \in \mathbf{N}}$. Let

$$N = \sum_{t=1}^{\infty} (f(x_t) - \rho_t)^p.$$

Then since

since $v_k > 0, x_k = 1, x_i < 1$, and $\lambda < 1$.

Assume as a second case that $x_i = 1$. In this case,

$$\begin{aligned}
\lim_{\gamma \to 0} 1/q_\gamma &= \lim_{\gamma \to 0} \frac{\sum_{j=1}^{n} v_j \left(\frac{(1+\gamma)(1-\lambda_t+\gamma)}{(\lambda_t+\gamma)\gamma} \right)^{\frac{x_j}{1+2\gamma}}}{v_i \left(\frac{(1+\gamma)(1-\lambda_t+\gamma)}{(\lambda_t+\gamma)\gamma} \right)^{\frac{1}{1+2\gamma}}} \\
&= \lim_{\gamma \to 0} \frac{\sum_{j:x_j=1} v_j}{v_i} + \lim_{\gamma \to 0} \sum_{j:x_j<1} \frac{v_j}{v_i} \left(\frac{(1+\gamma)(1-\lambda_t+\gamma)}{(\lambda_t+\gamma)\gamma} \right)^{\frac{x_j-1}{1+2\gamma}} \\
&= \frac{\sum_{j:x_j=1} v_j}{v_i}.
\end{aligned}$$

Combining this with (B.2) and (B.1) yields the desired result. □

Now we are ready for the main result of this appendix.

Theorem 80: *Choose $m \in \mathbf{N}, \vec{x}_1, ..., \vec{x}_m \in [0,1]^n$, and $\rho_1, ..., \rho_m \in [0,1]$ such that there is a $\vec{\mu} \in [0,1]^n$ whose components sum to 1 such that for all $t \leq m$, $\rho_t = \vec{\mu} \cdot \vec{x}_t$. The sequence $\vec{v}_1, ..., \vec{v}_m$ of vectors obtained though the update rule for A_0 is well-defined, and finite.*

Proof: The proof proceeds by induction on the trial t with an induction hypothesis consisting of the statement of the theorem, restricted to a specific trial t, together with the fact that for each $i \leq n$, if $v_{t,i} = 0$ then $\mu_i = 0$.

The induction hypothesis is trivially satisfied for $\vec{v}_1 = (1/n, ..., 1/n)$.

For the induction step, choose $t \geq 1$. Assuming the induction hypothesis holds for t, we wish to establish that it holds for $t+1$. The case in which $\lambda_t = \rho_t$ and that in which $\rho_t < 1$ and $\lambda_t > 0$ are both trivial. The case in which $\rho_t = 1$ and $\lambda_t < 1$ is handled easily using Lemma 79, since in that case, $x_{t,i} < 1$ implies that $\mu_i = 0$. Finally, assume $\lambda_t = 0$. In this case, for each i such that $\mu_i \neq 0$, we have $v_{t,i} \neq 0$ (the induction hypothesis), and for each i such that $v_{t,i} \neq 0$, we have $x_{t,i} = 0$ (since $\lambda_t = 0$). Thus for each i such that $\mu_i \neq 0$, we have $x_{t,i} = 0$, which implies that $\rho_t = 0$. The theorem follows trivially when $\lambda_t = \rho_t$.

This completes the proof. □

Appendix B. The finiteness of A_0's weights

In this section, we prove that the update

$$v_{t+1,i} = \lim_{\gamma \to 0} \frac{v_{t,i} \left(\frac{(\rho_t+\gamma)(1-\lambda_t+\gamma)}{(\lambda_t+\gamma)(1-\rho_t+\gamma)} \right)^{\frac{x_{t,i}}{1+2\gamma}}}{\sum_{j=1}^{n} v_{t,j} \left(\frac{(\rho_t+\gamma)(1-\lambda_t+\gamma)}{(\lambda_t+\gamma)(1-\rho_t+\gamma)} \right)^{\frac{x_{t,j}}{1+2\gamma}}}$$

used by A_0 preserves the finiteness of A_0's weights if there is a probability vector $\vec{\mu}$ such that for all $t, \rho_t = \vec{\mu} \cdot \vec{x}_t$. We begin with the following lemma.

Lemma 79: *Let $\vec{v} \in [0,1]$ have components which sum to 1. If $0 \leq \lambda < 1, \vec{x} \in [0,1]^n$ are such that is a $\vec{\mu} \in [0,1]^n$ whose components sum to 1 such that $\vec{\mu} \cdot \vec{x} = 1$ and for which $v_i = 0$ only if $\mu_i = 0$, then for each $i \in \mathbf{N}, i \leq n$*

$$\lim_{\gamma \to 0} \frac{v_i \left(\frac{(1+\gamma)(1-\lambda_t+\gamma)}{(\lambda_t+\gamma)(+\gamma)} \right)^{\frac{x_i}{1+2\gamma}}}{\sum_{j=1}^{n} v_j \left(\frac{(1+\gamma)(1-\lambda_t+\gamma)}{(\lambda_t+\gamma)\gamma} \right)^{\frac{x_j}{1+2\gamma}}} = \begin{cases} \frac{v_i}{\sum_{j: x_j = 1} v_j} & \text{if } x_i = 1 \\ 0 & \text{otherwise.} \end{cases}$$

Proof: Choose $i \leq \mathbf{N}$. Assume without loss of generality that $v_i > 0$. Since $\vec{x} \in [0,1]^n$, and $\vec{\mu} \in [0,1]^n$ has components which sum to 1, and $\vec{\mu} \cdot \vec{x} = 1$, there exists a k such that $x_k = 1$ and $\mu_k > 0$. Therefore, by assumption, $v_k > 0$ as well. For each $\gamma > 0$, let

$$q_\gamma = \frac{v_i \left(\frac{(1+\gamma)(1-\lambda_t+\gamma)}{(\lambda_t+\gamma)\gamma} \right)^{\frac{x_i}{1+2\gamma}}}{\sum_{j=1}^{n} v_j \left(\frac{(1+\gamma)(1-\lambda_t+\gamma)}{(\lambda_t+\gamma)\gamma} \right)^{\frac{x_j}{1+2\gamma}}}.$$

Then

$$\lim_{\gamma \to 0} q_\gamma = \frac{1}{\lim_{\gamma \to 0} 1/q_\gamma}. \tag{B.1}$$

Assume as a first case that $x_i < 1$. Then

$$\begin{aligned} \lim_{\gamma \to 0} 1/q_\gamma &= \lim_{\gamma \to 0} \frac{\sum_{j=1}^{n} v_j \left(\frac{(1+\gamma)(1-\lambda_t+\gamma)}{(\lambda_t+\gamma)\gamma} \right)^{\frac{x_j}{1+2\gamma}}}{v_i \left(\frac{(1+\gamma)(1-\lambda_t+\gamma)}{(\lambda_t+\gamma)\gamma} \right)^{\frac{x_i}{1+2\gamma}}} \\ &\geq \lim_{\gamma \to 0} \frac{v_k \left(\frac{(1+\gamma)(1-\lambda_t+\gamma)}{(\lambda_t+\gamma)\gamma} \right)^{\frac{x_k}{1+2\gamma}}}{v_i \left(\frac{(1+\gamma)(1-\lambda_t+\gamma)}{(\lambda_t+\gamma)\gamma} \right)^{\frac{x_i}{1+2\gamma}}} \\ &= \lim_{\gamma \to 0} \frac{v_k}{v_i} \left(\frac{(1+\gamma)(1-\lambda_t+\gamma)}{(\lambda_t+\gamma)\gamma} \right)^{\frac{x_k - x_i}{1+2\gamma}} \\ &= \infty \end{aligned} \tag{B.2}$$

$$\begin{aligned} x = y &\Leftrightarrow d(x,y) = 0 \\ d(x,y) &= d(y,x) \\ d(x,z) &\leq d(x,y) + d(y,z). \end{aligned}$$

In this case, we say (S,d) is a metric space. Let $T \subseteq S$. We say T is bounded if $\sup\{d(x,y) : x,y \in T\}$ is finite.

Appendix A. Mathematical Preliminaries

Throughout, we let $\mathbf{R}$ represent the real numbers, $\mathbf{R}^+$ represent the positive reals, $\mathbf{Q}$ represent the rationals, $\mathbf{N}$ represent the positive integers, $\mathbf{Z}$ denote the integers, and $\mathbf{Z}^+$ represent the nonnegative integers. Also, log always represents the base 2 logarithm, and ln represents the natural logarithm.

For $\vec{x} = (x_1, ..., x_n) \in \mathbf{R}^n$, and $p \in \mathbf{N}$,

$$||\vec{x}||_p = \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p}.$$

In particular,

$$\begin{aligned} ||\vec{x}||_1 &= \sum_{i=1}^{n} |x_i| \\ ||\vec{x}||_2 &= \sqrt{\sum_{i=1}^{n} x_i^2} = \sqrt{x \cdot x}. \end{aligned}$$

Also,

$$||\vec{x}||_\infty = \max_i x_i.$$

Recall that for a function $f : [0,1] \to \mathbf{R}$, and $q \geq 1$, the q-norm of f, denoted by $||f||_q$, is defined to be

$$\left(\int_{x=0}^{1} |f(x)|^q dx \right)^{1/q},$$

and

$$||f||_\infty = \lim_{q \to \infty} ||f||_q.$$

If X is a set, and D is a probability distribution on X, and if $\phi(x)$ is some mathematical statement containing x as a free variable, define $\mathbf{Pr}_{x \in D}(\phi(x))$ as $D(\{x \in X : \phi(x)\})$. Define $\mathbf{E}_{x \in D}$ similarly for expectations of random variables defined on X. We will drop the subscripts where there is no possibility of confusion.

Now, let S be a set. Let $d : S \times S \to \mathbf{R}^+$. We say that d is a metric on S if for all $x, y, z \in S$,

[Maass and Turan, 1989] W. Maass and G. Turan. On the complexity of learning from counterexamples. *Proceedings of the 30th Annual Symposium on the Foundations of Computer Science*, 1989.

[Maass and Turan, 1990] W. Maass and G. Turan. On the complexity of learning from counterexamples and membership queries. *Proceedings of the 31st Annual Symposium on the Foundations of Computer Science*, 1990.

[Maass, 1991] W. Maass. On-line learning with an oblivious environment and the power of randomization. *The 1991 Workshop on Computational Learning Theory*, pages 167–175, 1991.

[Mycielski, 1988] J. Mycielski. A learning algorithm for linear operators. *Proceedings of the American Mathematical Society*, 103(2):547–550, 1988.

[Natarajan, 1989] B.K. Natarajan. On learning sets and functions. *Machine Learning*, 4:67–97, 1989.

[Pitt and Valiant, 1988] L. Pitt and L.G. Valiant. Computational limitations on learning from examples. *Journal of the Association for Computing Machinery*, 35(4):965–984, 1988.

[Pitt and Warmuth, 1990] L. Pitt and M.K. Warmuth. Prediction preserving reducibility. *Journal of Computer and System Sciences*, 41(3), 1990.

[Pollard, 1984] D. Pollard. *Convergence of Stochastic Processes*. Springer Verlag, 1984.

[Pollard, 1990] D. Pollard. *Empirical Processes : Theory and Applications*. Institute of Mathematical Statistics, 1990.

[Sauer, 1972] N. Sauer. On the density of families of sets. *J. Combinatorial Theory (A)*, 13:145–147, 1972.

[Steele, 1978] J.M. Steele. Existence of submatrices with all possible columns. *Journal of Combinatorial Theory, Series A*, 24:84–88, 1978.

[Tomasta, 1981] P. Tomasta. Dart calculus of induced subsets. *Discrete Mathematics*, 34:195–198, 1981.

[Valiant, 1984] L.G. Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.

[Vapnik and Chervonenkis, 1971] V.N. Vapnik and A.Y. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and its Applications*, 16(2):264–280, 1971.

[Vapnik, 1982] V.N. Vapnik. *Estimation of Dependencies based on Empirical Data*. Springer Verlag, 1982.

[Vapnik, 1989] V.N. Vapnik. Inductive principles of the search for empirical dependences (methods based on weak convergence of probability measures). *The 1989 Workshop on Computational Learning Theory*, 1989.

[Widrow and Hoff, 1960] B. Widrow and M.E. Hoff. Adaptive switching circuits. *1960 IRE WESCON Conv. Record*, pages 96–104, 1960.

[Haussler, 1989b] D. Haussler. Learning conjunctive concepts in structural domains. *Machine Learning*, 4(1):7–40, 1989.

[Haussler, 1991] D. Haussler. Decision theoretic generalizations of the PAC model for neural net and other learning applications. Technical Report UCSC-CRL-91-02, University of California at Santa Cruz, 1991.

[Helmbold and Long, 1991] D.P. Helmbold and P.M. Long. Tracking drifting concepts using random examples. *The 1991 Workshop on Computational Learning Theory*, pages 13–23, 1991.

[Helmbold *et al.*, 1990] D. Helmbold, R. Sloan, and M.K. Warmuth. Learning integer lattices. *The 1990 Workshop on Computational Learning Theory*, pages 288–302, 1990.

[Karpovsky and Milman, 1978] M.G. Karpovsky and V.D. Milman. Coordinate density of sets of vectors. *Discrete Mathematics*, 24:177–184, 1978.

[Kearns and Li, 1988] M. Kearns and M. Li. Learning in the presence of malicious errors. *Proceedings of the 20th ACM Symposium on the Theory of Computation*, pages 267–279, 1988.

[Kearns *et al.*, 1987] M. Kearns, M. Li, L. Pitt, and L.G. Valiant. On the learnability of boolean formulae. *Proceedings of the 19th Annual Symposium on the Theory of Computation*, pages 285–295, 1987.

[Kimber and Long, 1992] D. Kimber and P.M. Long. The learning complexity of smooth functions of a single variable. To appear, *The 1992 Workshop on Computational Learning Theory*, 1992.

[Kuh *et al.*, 1991] A. Kuh, T. Pesche, and R. Rivest. Lower bounds on mistake rates for incremental learning algorithms when concepts drift. Unpublished manuscript, 1991.

[Kullback, 1967] S. Kullback. A lower bound for discrimination in terms of variation. *IEEE transactions on Information Theory*, 13:126–127, 1967.

[Leitmann, 1981] G. Leitmann. *The Calculus of Variations and Optimal Control*. Plenum Press, 1981.

[Littlestone and Warmuth, 1989] N. Littlestone and M.K. Warmuth. The weighted majority algorithm. *Proceedings of the 30th Annual Symposium on the Foundations of Computer Science*, 1989.

[Littlestone *et al.*, 1991] N. Littlestone, P.M. Long, and M.K. Warmuth. On-line learning of linear functions. *Proceedings of the 23rd ACM Symposium on the Theory of Computation*, pages 465–475, 1991.

[Littlestone, 1988] N. Littlestone. Learning quickly when irrelevant attributes abound: a new linear-threshold algorithm. *Machine Learning*, 2:285–318, 1988.

[Littlestone, 1989a] N. Littlestone. From on-line to batch learning. *The 1989 Workshop on Computational Learning Theory*, pages 269–284, 1989.

[Littlestone, 1989b] N. Littlestone. *Mistake Bounds and Logarithmic Linear-threshold Learning Algorithms*. PhD thesis, UC Santa Cruz, 1989.

[Littlestone, 1991] N. Littlestone, 1991. Personal communication.

[Blum, 1990c] A. Blum. Separating PAC and mistake-bound learning models over the boolean domain. *Proceedings of the 31st Annual Symposium on the Foundations of Computer Science*, 1990.

[Blumer *et al.*, 1989] A. Blumer, A. Ehrenfeucht, D. Haussler, and M.K. Warmuth. Learnability and the Vapnik-Chervonenkis dimension. *JACM*, 36(4):929–965, 1989.

[Bondy, 1972] J.A. Bondy. Induced subsets. *Journal of Combinatorial Theory (B)*, 12:201–202, 1972.

[Cesa-Bianchi *et al.*, 1991] N. Cesa-Bianchi, P.M. Long, and M.K. Warmuth. A comparison of on-line algorithms for learning linear functions. Manuscript, 1991.

[Duda and Hart, 1973] R.O. Duda and P.E. Hart. *Pattern Recognition and Scene Analysis*. John Wiley and Sons, 1973.

[Dudley, 1984] R.M. Dudley. A course on empirical processes. *Lecture notes in mathematics*, 1097:2–142, 1984.

[Dudley, 1987] R.M. Dudley. Universal donsker classes and metric entropy. *Ann. Prob.*, 15(4):1306–1326, 1987.

[Ehrenfeucht *et al.*, 1989] A. Ehrenfeucht, D. Haussler, M. Kearns, and L.G. Valiant. A general lower bound on the number of examples needed for learning. *Information and Computation*, 82(3):247–251, 1989.

[Faber and Mycielski, 1991] V. Faber and J. Mycielski. Applications of learning theorems. *Fundamenta Informaticae*, 15(2):145–167, 1991.

[Frankl *et al.*, 1987] P. Frankl, Z. Furedi, and J. Pach. Bounding one-way differences. *European Journal of Combinatorics*, 3:341–347, 1987.

[Frankl, 1983] P. Frankl. On the trace of finite sets. *Journal of Combinatorial Theory (A)*, 34:41–45, 1983.

[Hardle, 1991] W. Hardle. *Smoothing Techniques*. Springer Verlag, 1991.

[Haussler and Long, 1990] D. Haussler and P.M. Long. A generalization of Sauer's lemma. Technical Report UCSC-CRL-90-15, University of California at Santa Cruz, 1990.

[Haussler *et al.*, 1988] D. Haussler, N. Littlestone, and M.K. Warmuth. Predicting $\{0,1\}$ functions on randomly drawn points. *Proceedings of the 29th Annual Symposium on the Foundations of Computer Science*, pages 100–109, 1988.

[Haussler *et al.*, 1990] D. Haussler, N. Littlestone, and M.K. Warmuth. Predicting $\{0,1\}$ functions on randomly drawn points. Technical report, University of California at Santa Cruz, 1990.

[Haussler *et al.*, 1991] D. Haussler, M. Kearns, and R. Schapire. Bounds on the sample complexity of bayesian learning using information theory and the VC-dimension. *The 1991 Workshop on Computational Learning Theory*, pages 61–74, 1991.

[Haussler, 1988] D. Haussler. Space efficient learning algorithms. Technical report, UC Santa Cruz, 1988.

[Haussler, 1989a] D. Haussler. Generalizing the PAC model: sample size bounds from metric dimension-based uniform convergence results. *Proceedings of the 30th Annual Symposium on the Foundations of Computer Science*, 1989.

References

[Aldous and Vazirani, 1990] D. Aldous and U. Vazirani. A Markovian extension of Valiant's learning model. *Proceedings of the 31st Annual Symposium on the Foundations of Computer Science*, pages 392–396, 1990.

[Alon, 1983] N. Alon. On the density of sets of vectors. *Discrete Mathematics*, 24:177–184, 1983.

[Angluin and Valiant, 1979] D. Angluin and L. Valiant. Fast probabilistic algorithms for Hamiltonion circuits and matchings. *Journal of Computer and System Sciences*, 18(2):155–193, 1979.

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

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

[Anstee and Furedi, 1986] R.P. Anstee and Z. Furedi. Forbidden submatrices. *Discrete Math*, 62:225–243, 1986.

[Anstee, 1985] R.P. Anstee. General forbidden configuration theorems. *Journal of Combinatorial Theory (A)*, 40:108–124, 1985.

[Anstee, 1988] R.P. Anstee. A forbidden configuration theorem of Alon. *Journal of Combinatorial Theory (A)*, 47:16–27, 1988.

[Anstee, 1991] R.P. Anstee. On a conjecture concerning forbidden submatrices. Unpublished manuscript, 1991.

[Anthony *et al.*, 1990] M. Anthony, N. Biggs, and J. Shawe-Taylor. The learnability of formal concepts. *The 1990 Workshop on Computational Learning Theory*, pages 246–257, 1990.

[Barron, 1991] A. Barron. Approximation and estimation bounds for artificial neural networks. *The 1991 Workshop on Computational Learning Theory*, 1991.

[Ben-David *et al.*, 1992] S. Ben-David, N. Cesa-Bianchi, and P.M. Long. Characterizations of learnability for classes of $\{0, ..., n\}$-valued functions. To appear, *The 1992 Workshop on Computational Learning Theory*, 1992.

[Bernstein, 1992] E.J. Bernstein. Absolute error bounds for learning linear functions on line. To appear in the *Proceedings of the 1992 Workshop on Computational Learning Theory*, 1992.

[Blum *et al.*, 1991] A. Blum, L. Hellerstein, and N. Littlestone. Learning in the presence of finitely many or infinitely many irrelevant attributes. *The 1991 Workshop on Computational Learning Theory*, 1991.

[Blum, 1990a] A. Blum. Learning boolean functions in an infinite attribute space. *Proceedings of the 22nd ACM Symposium on the Theory of Computation*, 1990.

[Blum, 1990b] A. Blum. Separating PAC and mistake-bound learning models over the boolean domain. *Proceedings of the 31st Annual Symposium on the Foundations of Computer Science*, 1990.

and instead of wanting to make the probability of mistake small, we want to make the expectation of the absolute value of the difference between our prediction and the truth small. In place of an algorithm for minimizing disagreements, we require an algorithm for minimizing the sum of absolute errors on a sample. It would be interesting to obtain results for more general loss functions, e.g. the square loss. Also, we have no general lower bounds for the tracking of real valued functions.

Other natural problems include: optimizing the constants and removing the $1/\ln\frac{1}{\epsilon}$ gap between our bounds on Δ.

to determine when an adjustment is required. It is often infeasible to inspect each item produced as the inspection process might be very expensive or even destroy the good. Thus a more complicated inspection plan indicating when to inspect and how to evaluate the inspection results is needed. The results in Section 7.2 are applicable to this problem.

Intuitively, the following approach seems as if it should lead to improved tracking algorithms. Instead of simply minimizing the number of disagreements with a suffix of the previous examples, an algorithm might weight previous examples with gradually decreasing nonnegative weights which sum to one. Then for each hypothesis h in the target class, the algorithm might use the sum of the weights of the examples with which h disagrees as the estimate of the probability that it will make a mistake on the next trial, then use the hypothesis which minimizes this, possibly more accurate, estimate. One wonders whether such an algorithm might significantly improve on the simple "minimize disagreements" algorithm analyzed in this chapter.

It is easy to see how to alter our arguments to obtain results in a related model (often called "agnostic learning") in which the algorithm doesn't know a priori a class which contains each of the sequence of targets, and tries to predict nearly as well as possible using hypotheses in a certain class $\mathcal{F}$. More formally, suppose for a worst case sequence of concepts $f_1, f_2, ...$ (not necessarily in the hypothesis class $\mathcal{F}$), for each t we defined κ_t to be $\min_{h \in \mathcal{F}} \mathbf{Pr}(h(x) \neq f_t(x))$. It can be shown by modifying the proofs of Section 7.2, that for $\Delta \leq c\epsilon^3/(d \ln(1/\epsilon))$, an algorithm can achieve probability of mistake at most $\kappa_t + \epsilon$ for all large enough t [Helmbold and Long, 1991]. One wonders whether these results can be improved.

Haussler [Haussler, 1991] has generalized the results of [Blumer *et al.*, 1989] to apply to learning in many frameworks, one of which is the learning of real valued functions. Using Haussler's results, the techniques of Section 7.2 can trivially be extended to apply to uniformly bounded classes of real valued functions (e.g., feed forward neural networks of a particular architecture which has one output node), where, in place of the Vapnik-Chervonenkis dimension, we use Pollard's *pseudo*-dimension [Pollard, 1984,Haussler, 1991],

Theorem 78: *For all $\epsilon < 1/e^2$ and $n \in \mathbf{N}$, $HALFSPACES_n$ is not (ϵ, Δ)-trackable when $\Delta > e^4\epsilon^2/n$, and $BOXES_n$ is not (ϵ, Δ)-trackable when $\Delta > e^4\epsilon^2/2n$.*

This theorem, along with the facts that the VC dimension of HALFSPACES_n is $n+1$ and that of and BOXES_n is $2n$, establishes that the general purpose algorithm described in Section 7.2 is within a constant times a log factor of optimal for these two natural concept classes.

7.5 Discussion

In this chapter, we have defined a learning model in which the target concept is allowed to change over time and discovered a general-purpose algorithm whose performance nearly matches our lower bounds (on at least two natural target classes). However this algorithm relies on a potentially expensive subroutine for minimizing disagreements within a constant factor. To combat this difficulty, we have found an efficient way to approximately minimize disagreements to within a factor that depends (linearly) on the VC-dimension. This gives us a second generic algorithm which, although not proven able to tolerate quite as much drift, is more likely to be computationally efficient (as it is for halfspaces, hyperrectangles, and any other target class which is properly PAC learnable).

Our algorithms are robust in the sense that they don't need to know the rate of drift Δ ahead of time, although attempting to achieve an accuracy ϵ amounts to an implicit assumption of an upper bound on Δ.

Although our results have usually been stated in terms of how much target motion can be tolerated, they can viewed in other ways. Bounds like "all $\Delta < c\epsilon^2/(d^2 \ln \epsilon)$ are tolerated" are easily converted to "the error rate, ϵ, is at most $c_\alpha d \Delta^{1/(2-\alpha)}$ for arbitrarily small α." Also, our bounds indicate how frequently one must sample to achieve a desired accuracy when given a bound on the continuous rate of target drift. This interpretation may be the more useful one.

Consider an assembly line process where the machines slowly drift out of alignment, gradually increasing the defect rate. One wants to sample the finished products in order

since, when given E, it is equally likely that $f_{\bar{z},t}(x_m)$ is 0 or 1, independent of the previous examples. Now,

$$\begin{aligned}
\mathbf{Pr}(E) &= \mathbf{Pr}\left(x_m - \frac{\lfloor nx_m \rfloor}{n} \leq \frac{t\Delta}{n} \;\&\; \forall 0 < i < t, x_{m-t+i} \notin \left[\frac{\lfloor nx_m \rfloor}{n}, \frac{\lfloor nx_m \rfloor}{n} + \frac{i\Delta}{n}\right]\right) \\
&= t\Delta \prod_{i=1}^{t-1}(1 - \frac{\Delta i}{n}) \\
&\geq t\Delta \prod_{i=1}^{t-1} \exp\left(\frac{-\frac{\Delta i}{n}}{1 - \frac{\Delta i}{n}}\right) \\
&= t\Delta \exp\left(\sum_{i=1}^{t-1} \frac{-\frac{\Delta i}{n}}{1 - \frac{\Delta i}{n}}\right) \\
&\geq t\Delta \exp\left(\left(\frac{-\frac{\Delta}{n}}{1 - \frac{\Delta t}{n}}\right)\frac{t^2}{2}\right) \\
&\geq t\Delta \exp\left(-\frac{e}{2(e-1)}\frac{t^2\Delta}{n}\right) \qquad \text{(since } t \leq n/(e\Delta)\text{)} \\
&\geq \frac{2}{3}\sqrt{n\Delta}\exp\left(-\frac{e}{2(e-1)}\right) \qquad \text{(since } \tfrac{2}{3}\sqrt{n/\Delta} \leq t \leq \sqrt{n/\Delta}\text{)}
\end{aligned}$$

Noting that $\frac{2}{3}\exp\left(-\frac{e}{2(e-1)}\right) > \frac{2}{e^2}$ yields

$$\begin{aligned}
\mathbf{Pr}(\text{mistake}) &> \frac{\sqrt{n\Delta}}{e^2} \\
&> \epsilon.
\end{aligned}$$

Since

$$\mathbf{Pr}_{(\bar{x},\bar{z},\sigma)\in U^{m+1}\times U'\times\mathcal{U}}(\text{mistake}(\bar{z},\bar{x},\sigma)) > \epsilon,$$

there is a $\bar{z}$ for which

$$\mathbf{Pr}_{(\bar{x},\sigma)\in U^{m+1}\times\mathcal{U}}(\text{mistake}(\bar{z},\bar{x},\sigma)) > \epsilon,$$

contradicting the assumption that that A (ϵ,Δ)-tracks BASIC_n.□

Recall the definitions of HALFSPACES_n and BOXES_n from the previous section.

The following theorem follows from the bounds for BASIC_n via a trivial embedding of BASIC_n into HALFSPACES_n and a similar embedding of BASIC_{2n} into BOXES_n using a simplified version of prediction preserving reductions [Pitt and Warmuth, 1990]. The same embeddings were employed in [Haussler *et al.*, 1990]. The details are omitted.

Theorem 77: *For all $n \in \mathbf{N}$, $BASIC_n$ is not (ϵ, Δ)-trackable if $\epsilon \leq 1/e^2$ and $\Delta \geq e^4\epsilon^2/n$.*

Proof: By contradiction. Assume that tracking strategy A (ϵ, Δ)-tracks BASIC_n for some $0 < \epsilon \leq 1/e^2$, $n \in \mathbf{N}$, and $\Delta \geq e^4\epsilon^2/n$. Thus after seeing at least m_0 examples drawn from distribution D and labeled by any (Δ, D)-admissible sequence of targets, the probability that A makes a mistake on the next example is at most ϵ.

Without loss of generality, set $\Delta = e^4\epsilon^2/n$. With the restriction on ϵ, $\Delta \leq 1/n$ (and $n \leq 1/\Delta$). Also, since no non-degenerate class is (ϵ, Δ)-trackable if $\Delta > \epsilon$ and $\epsilon \leq 1/3$, we may assume that $\Delta \leq 1/e^2$.

Let $t = \lfloor \sqrt{n/\Delta} \rfloor$. Since $e \leq \sqrt{e^2 n} \leq \sqrt{n/\Delta}$, we get $\frac{2}{3}\sqrt{n/\Delta} \leq t \leq \sqrt{n/\Delta}$ and $et \leq n/\Delta$. These inequalities will be used at the end of the proof.

For each $\bar{z} \in \{0,1\}^n$ and $0 \leq i \leq t$, define $f_{\bar{z},i} \in \mathrm{BASIC}_n$ as the indicator function for

$$\cup_{j=1}^{n} [j/n, (j + i\Delta z_j)/n).$$

Since $t \leq 1/\Delta$ (using $n \leq 1/\Delta$), every interval in the union has length at most $1/n$. Note that $f_{\bar{z},0}$ is the function mapping everything to 0. Choose m such that $m \geq t+1$ and $m \geq m_0$. Let $S(\bar{z})$ be the sequence of m elements of BASIC_n defined by $S(\bar{z}) = (f_{\bar{z},0}, f_{\bar{z},0}, \ldots, f_{\bar{z},0}, f_{\bar{z},1}, f_{\bar{z},2}, \ldots, f_{\bar{z},t})$. Let U be the uniform distribution on $X = [0,1]$. One can easily verify that for all $\bar{z} \in \{0,1\}^n$, $S(\bar{z})$ is (Δ, U)-admissible.

Let E be the event that for a random $\bar{x} \in [0,1]^m$, x_m is the first "passed" point in its subinterval. More formally, $x_m - \frac{\lfloor n x_m \rfloor}{n} \leq \frac{t\Delta}{n}$ and for all $0 < i < t$, $x_{m-t+i} \notin \left[\frac{\lfloor n x_m \rfloor}{n}, \frac{\lfloor n x_m \rfloor}{n} + \frac{i\Delta}{n}\right]$.

For each $\bar{z} \in \{0,1\}^n, \bar{x} \in [0,1]^m$, $\sigma \in \Gamma$, let mistake$(\bar{z}, \bar{x}, \sigma)$ be the event that

$$A(\mathrm{sam}_{m-1}(S(\bar{z}), \bar{x}), x_m, \sigma) \neq f_{\bar{z},t}(x_m),$$

i.e. that strategy A incorrectly predicts the label of the mth example where σ represents the strategy's internal randomization. Finally, let U' be the uniform distribution over $\{0,1\}^n$. We have

$$\begin{aligned}&\mathbf{Pr}_{(\bar{x},\bar{z},\sigma) \in U^m \times U' \times \mathcal{U}}(\mathrm{mistake}(\bar{z}, \bar{x}, \sigma)) \\ &\quad \geq \mathbf{Pr}(\mathrm{mistake}(\bar{z}, \bar{x}, \sigma) | E)\mathbf{Pr}(E) = \tfrac{1}{2}\mathbf{Pr}(E)\end{aligned}$$

7.4 Upper bounds on the tolerable amount of drift

In this section we prove upper bounds on the tolerable amount of drift for two commonly studied concept classes: halfspaces and axis-aligned rectangles. Our upper bounds show that the algorithm of Section 7.2 is within a log times a constant factor of optimal for each of these classes.

First, we will prove an upper bound for BASIC_n, the class of indicator functions for the following family of subsets of the unit interval:

$$\{\cup_{i=1}^{n}[i/n,(i+a_i)/n) : \bar{a} \in [0,1]^n\}.$$

This class can be viewed as dividing the unit interval into n subintervals of equal length. Every concept in the class is the union of an initial segment from each of the subintervals. It is easy to see that $VCdim(\mathrm{BASIC}_n) = n$.

Our argument for the upper bound on BASIC_n uses ideas from earlier arguments giving lower bounds on the probability of a mistake when predicting a stationary target function [Ehrenfeucht *et al.*, 1989] [Haussler *et al.*, 1990].

The intuition behind the argument is as follows. Suppose there is a water truck rolling down a section of dusty road at 10 kilometers per hour. Either the truck is empty or it is spraying water (unknown to us, but both possibilities are equally likely). Each minute a point on the road is picked at random and we predict whether or not the point is wet before looking at it. If the point has not yet been passed by the water truck, then we can safely predict that it is dry. If a previously picked point had already been passed by the water truck when it was picked, then we know whether or not the truck is spraying water and can always predict correctly. However, our prediction always has a 1/2 chance of being wrong the first time we see a point that the water truck has passed. This idea can be extended to to n watertrucks (each of which is independently spraying or empty) on n different roads. Whenever a point on road i that has been passed by truck i is picked, and none of the previous points had been passed by truck i when they were picked, we will make a mistake with probability 1/2.

Theorem 73 ([Pitt and Valiant, 1988]): *If $\mathcal{F} \subseteq \cup_n 2^{\mathbf{Q}^n}$ is properly PAC learnable, then there is a randomized polynomial time algorithm which solves the consistency problem for $\mathcal{F}$.*

Theorem 74 ([Blumer *et al.*, 1989]): *If $\mathcal{F} = \cup_n \mathcal{F}_n$, where $\mathcal{F}_n \subseteq \mathbf{Q}^n$ is properly PAC learnable, then there is a polynomial p such that for all $n \in \mathbf{N}$, $VCdim(\mathcal{F}_n) \leq p(n)$.*

Combining these with Corollary 72 we obtain the following.

Corollary 75: *Let $\mathcal{F}$ be a stratified tracking problem. Then if the corresponding learning problem is properly PAC learnable, $\mathcal{F}$ is efficiently trackable.*

Combining Corollary 72 with Theorem 69, we obtain the following result for halfspaces and hyperrectangles in particular. Let HALFSPACES$_n$ be the set of indicator functions for the following sets:

$$\{\{\vec{x} \in \mathbf{Q}^n : \vec{a} \cdot \vec{x} \geq b\} : \vec{a} \in \mathbf{Q}^n, b \in \mathbf{Q}\}.$$

Let BOXES$_n$ be the set of indicator functions for the set of axis parallel hyperrectangles in n-dimensional space, i.e.

$$\{\prod_{i=1}^{n} [a_i, b_i] : \vec{a}, \vec{b} \in \mathbf{Q}^n\}.$$

Corollary 76: *There is a constant $c > 0$ and there are efficient tracking algorithms for each of $\{HALFSPACES_n : n \in \mathbf{N}\}$ and $\{BOXES_n : n \in \mathbf{N}\}$ that (ϵ, Δ)-track these classes for*

$$\Delta \leq \frac{c\epsilon^2}{n^2 \log(1/\epsilon)}.$$

Finally, Kearns and Li [Kearns and Li, 1988] showed that, loosely speaking, significantly improving the factor of approximation of our algorithm for minimizing disagreements for hyperrectangles (in particular, removing the dependence on d) would lead to corresponding improvements on the approximation algorithm for set cover, which has not been significantly improved since the 1970's. Nevertheless, it remains possible that, via other methods, one might obtain efficient algorithms that tracks this class at rates closer to optimal.

$$\begin{aligned} &\geq \frac{2q-1}{2}\exp\left(\frac{-opt\,s}{m-opt}\right) \\ &\geq \frac{2q-1}{2}\exp\left(\frac{-opt}{m-opt}\right)\exp\left(\frac{-d}{\gamma}\ln\frac{em}{2d}\right) \\ &\geq \frac{2q-1}{2}e^{-1/\gamma}\left(\frac{2d}{em}\right)^{\frac{d}{\gamma}}. \end{aligned}$$

Thus, the probability that the hypothesis returned after l iterations has more than $(\gamma+1)opt$ disagreements with S is at most

$$\left(1-\frac{2q-1}{2e^{1/\gamma}}\left(\frac{2d}{em}\right)^{\frac{d}{\gamma}}\right)^{l} \leq \exp\left(-l\frac{2q-1}{2e^{1/\gamma}}\left(\frac{2d}{em}\right)^{\frac{d}{\gamma}}\right)$$

This completes the proof. □

By appropriate choice of γ and l, we obtain the following.

Corollary 72: *If $\gamma = d$ and $l \geq \frac{e^{1+1/d}m}{d(2q-1)}\ln\frac{1}{\delta}$ then with probability at least $1-\delta$ Algorithm Min-Disagreements returns a hypothesis consistent with all but $(\gamma+1)opt$ of the examples in S.*

Proof: If $opt = 0$, then the corollary is trivial. Assume $opt \geq 1$. Then

$$\begin{aligned} \mathbf{Pr}(\textit{algorithm fails}) &\leq \exp\left(-l\frac{2q-1}{2e^{1/\gamma}}\left(\frac{2d}{em}\right)^{\frac{d}{\gamma}}\right) \\ &= \exp\left(-ld\frac{2q-1}{me^{1+1/d}}\right) \\ &\leq \delta. \end{aligned}$$

This completes the proof. □

Note that if d and m grow polynomially with n, then the number l of iterations required by the algorithm in the previous corollary is also polynomial in n.

We may similarly show that, in fact, that for any $c > 0$, we can approximately minimize disagreements to within a factor of $cVCdim(\mathcal{F}_n)+1$ in $\text{poly}(m,n)^{1/c}$ time, if there is a polynomial time randomized algorithm for $\mathcal{F}$'s consistency problem and $VCdim(F_n)$ grows polynomially in n.

We can now take advantage of the following two theorems, which address learning in Valiant's PAC model [Valiant, 1984].

Consider the stage of the algorithm where $\widehat{opt} = opt$ and a particular iteration j of the inner loop where $\mathcal{A}$ produces hypothesis h'. Let *clean* be the event that none of the examples sampled during iteration j are in *bad* and *consist* be the event that h' is consistent with the subsample. By applying a standard approximation, we have

$$\begin{aligned}\mathbf{Pr}(clean \text{ and } consist) &\geq q(1 - opt/m)^s \\ &\geq q \exp\left(\frac{-opt\, s}{m - opt}\right).\end{aligned}$$

Now define *close* to be the event that h' agrees with all but $\gamma\ opt$ of the examples in $S - bad$, i.e. $\mathbf{Pr}_{z \in D'}(h'(z) \neq h_{opt}(z)) \leq \gamma\ opt/(m - opt)$. (Note that when *close* occurs, h' agrees with all but $(\gamma + 1)opt$ of the examples in S.) We have

$$\begin{aligned}&\mathbf{Pr}_{(S',\sigma) \in D^s \times \mathcal{U}}(\overline{close} \mid clean \text{ and } consist) \\ &= \mathbf{Pr}_{(S',\sigma) \in (D')^s \times \mathcal{U}}(\overline{close} \mid consist)\end{aligned} \tag{7.2}$$

since the distribution obtained by conditioning D^s on *clean* is $(D')^s$ (recall that $\mathcal{U}$ is the uniform distribution over sequences of bits, so that σ represents the randomization of consistency algorithm $\mathcal{A}$). Note that if both *clean* and *consist* occur then h' and h_{opt} agree with the examples in the subsample. Thus,

$$\begin{aligned}&\mathbf{Pr}_{(S',\sigma) \in D^s \times \mathcal{U}}(\overline{close} \mid clean \text{ and } consist) \\ &\leq \mathbf{Pr}_{(S',\sigma) \in (D')^s \times \mathcal{U}}(\overline{close} \text{ and } consist)/\mathbf{Pr}(consist) \\ &\leq \tfrac{1}{q}\mathbf{Pr}_{(S',\sigma) \in (D')^s \times \mathcal{U}}(\mathbf{Pr}_{z \in D'}(h'(z) \neq h_{opt}(z)) > \gamma\ opt/(m - opt) \\ &\qquad\qquad \text{and } \forall (x, y) \in S', h'(x) = h_{opt}(x)) \\ &\leq 1/2q,\end{aligned} \tag{7.3}$$

where the last inequality follows from Theorems 32 and 70 and the algorithm's choice of s. Thus,

$$\mathbf{Pr}_{(S',\sigma) \in D^s \times \mathcal{U}}(close \mid clean \text{ and } consist) \geq (2q - 1)/2q.$$

Now we can bound the probability of *close*.

$$\begin{aligned}\mathbf{Pr}_{(S',\sigma) \in D^s \times \mathcal{U}}(close) &\geq \mathbf{Pr}(close \text{ and } clean \text{ and } consist) \\ &= \mathbf{Pr}(close \mid clean \text{ and } consist)\mathbf{Pr}(clean \text{ and } consist)\end{aligned}$$

Algorithm Min-Disagreements

Inputs: a sample S of m examples; l, the number of iterations to run; $d = VCdim(\mathcal{F}_n)$;
and desired approximation factor $\gamma > 1$.

Uses: A randomized algorithm $\mathcal{A}$ for the consistency problem associated with $\mathcal{F}_n$.

choose an $h \in \mathcal{F}_n$ arbitrarily
for $j := 1$ to ld do
    run $\mathcal{A}$ on S' obtaining hypothesis h';
    if h' is consistent with S then stop and return h'
end for;
for $\widehat{opt} := 1$ to m/γ do
    $s := \left\lceil (d(m - \widehat{opt})/\gamma\widehat{opt}) \ln \frac{em}{2d}) \right\rceil$
    for $j := 1$ to l do
        draw S', an s-element subsample of S uniformly at random with replacement;
        run $\mathcal{A}$ on S' obtaining hypothesis h';
        if h' has fewer disagreements with S than h, set $h := h'$;
    end for;
end for;
return h;

Figure 7.1: Algorithm Min-Disagreements

the minimum possible number of disagreements between the sample and an $h \in \mathcal{F}$. We focus our attention on the case where $opt < m/(\gamma + 1)$, since otherwise the theorem is trivial as any hypothesis is consistent with all but $(\gamma + 1)opt$ examples of S.

Choose h_{opt} from among those hypotheses in $\mathcal{F}_n$ which have opt disagreements with S. Let $bad \subseteq S$ be the subset of the examples in S with which h_{opt} disagrees. Let D be the uniform distribution over S, and let D' be the uniform distribution over $S - bad$.

First, we will make use of the following observation of Vapnik's [Vapnik, 1982].

Theorem 70 ([Vapnik, 1982]): *Let X be a set and let $\mathcal{F}$ be a finite concept class over X. Let D be a probability distribution over X. Choose $f \in \mathcal{F}$ and $\epsilon < 1/2$. Then if $s \geq \frac{\ln(|\mathcal{F}|/2)}{\epsilon}$,*

$$\mathbf{Pr}_{\vec{x} \in D^s}(\exists h \in \mathcal{F} : \forall i, h(x_i) = f(x_i) \text{ and } \mathbf{Pr}_{y \in D}(h(y) \neq f(y)) \geq \epsilon) \leq 1/2.$$

Now, we turn to the main result of this section. If $\mathcal{F} = \cup_n \mathcal{F}_n$ is a stratified tracking problem, then the consistency problem associated with $\mathcal{F}$ is as follows:

> Given a sample in $2^{\mathbf{Q}^n \times \{0,1\}}$, find any hypothesis in $\mathcal{F}_n$ consistent with the sample if there is one, otherwise return any $h \in \mathcal{F}_n$.

A randomized polynomial time algorithm for the consistency problem returns, in time polynomial in n and the size of the sample, an h in $\mathcal{F}_n$. If the sample is consistent with some hypothesis in $\mathcal{F}_n$ then, with probability $q > 1/2$, the returned h will be consistent with the sample. Note that by repeatedly running such an algorithm (and checking each result against the sample) an arbitrarily high confidence can be acheived.

Algorithm Min-Disagreements (see Figure 7.1) uses a randomized polynomial time algorithm for the consistency problem to approximately minimize the number of disagreements.

It should be obvious that if $\mathcal{A}$ runs in randomized polynomial time then the algorithm Min-Disagreements runs in time polynomial in n,d, l and m.

Theorem 71: *For any $n \in \mathbf{N}$, $\mathcal{F}_n \subseteq 2^{\mathbf{Q}^n}$ of VC-dimension d, and set of m examples S, if $\mathcal{A}$ solves $\mathcal{F}_n$'s consistency problem with probability $q > 1/2$ and there is an element of $\mathcal{F}_n$ consistent with all but opt of the examples in S, then Algorithm Min-Disagreements with inputs S,m,l,d,γ finds a hypothesis consistent with all but $(\gamma + 1)opt$ examples in S with probability at least*

$$1 - \exp\left(-l\frac{2q-1}{2e^{1/\gamma}}\left(\frac{2d}{em}\right)^{\frac{d}{\gamma}}\right).$$

Proof: Choose $m \in \mathbf{N}$ and let $S = \{(x_i, y_i) : 1 \leq i \leq m\}$ be a sample. Let

$$opt = \min\{|\{i : h(x_i) \neq y_i\}| : h \in \mathcal{F}\},$$

the first m trials to within a factor of k. Choose ϵ and m as in Theorem 67. Then if $\Delta \leq \frac{\epsilon}{12k(m+1)}$, the probability that A makes a mistake on the $(m+1)$st trial of a (Δ, D)-admissible sequence of functions is at most ϵ.

Note that by ignoring (not counting disagreements with) examples beyond a certain point in the past we can, loosely speaking, make any later trial "look like" the $(m+1)$st trial. This observation leads to the following Corollary.

Corollary 69: *Let X be a domain, and $\mathcal{F}$ be a class of concepts over X of VC-dimension d. Assume A is a randomized algorithm which with probability $1-\epsilon/6$ finds an $h \in \mathcal{F}$ which approximates, to within a constant factor k, the minimum number of disagreements on a sample. Let A' be the tracking algorithm which predicts using the hypothesis produced by A from the most recent $m = \lceil (c_1 d/\epsilon)\log(1/\epsilon) \rceil$ examples, where $c_1 > 0$ depends on k. There is a positive constant c_2, depending only on k, such that for any $0 < \Delta < \epsilon$ where*

$$\Delta \leq \frac{c_2 \epsilon^2}{d \log \frac{1}{\epsilon}},$$

strategy A' (ϵ, Δ)-tracks $\mathcal{F}$.

7.3 Efficiently approximately minimizing disagreements

In this section we discuss the application of the techniques of [Kearns and Li, 1988] to the problem of approximately minimizing disagreements from among the hypotheses in a class $\mathcal{F}$, showing that if there is an efficient algorithm which returns a hypothesis with no disagreements if there is one, then there is an efficient randomized algorithm which with high probability returns a hypothesis that minimizes disagreements to within a factor of a constant times the VC-dimension of $\mathcal{F}$. Results very similar to those described here are implicit in the work of Kearns and Li (Theorems 12 and 16), although some minor modifications are necessary.[3] Also, we make use of the techniques of [Kearns and Li, 1988] in our proof.

[3]The difference between the result trivially obtainable by combining Theorems 12 and 16 of [Kearns and Li, 1988] and our result is that in the former, the sample is restricted to have the same number of positive and negative examples.

If *mistake* is the event that A makes a mistake on trial $m+1$, we have

$$\begin{aligned}
\mathbf{Pr}_{(\bar{x},y,\sigma)\in D^m\times D\times\mathcal{U}}(mistake) &\leq \mathbf{Pr}(mistake\cap\bar{E})+\mathbf{Pr}(mistake\cap E)\\
&\leq \mathbf{Pr}(mistake\cap\bar{E})+\mathbf{Pr}(E)\\
&\leq \mathbf{Pr}(mistake\cap\bar{E})+\epsilon/3\\
&\leq \mathbf{Pr}(mistake\cap\bar{E}\cap G)+\mathbf{Pr}(mistake\cap\bar{E}\cap\bar{G})+\epsilon/3\\
&\leq \mathbf{Pr}(\bar{E}\cap G)+2\epsilon/3. \qquad (7.1)
\end{aligned}$$

Next, we have

$$\begin{aligned}
\mathbf{Pr}(\bar{E}\cap G) &= \mathbf{Pr}\left(\mathbf{er}_{f_{m+1}}(h_{\bar{x},\sigma})>\epsilon/3 \text{ and } \frac{1}{m}\sum_{i=1}^{m} l(f_i(x_i),f_{m+1}(x_i))\leq\epsilon/(12k)\right.\\
&\qquad\qquad \left. \text{and } h_{\bar{x},\sigma}\in \text{mindis}(\bar{x})\right)\\
&\leq \mathbf{Pr}\left(\mathbf{er}_{f_{m+1}}(h_{\bar{x},\sigma})>\epsilon/3 \text{ and } \frac{1}{m}\sum_{i=1}^{m} l(f_i(x_i),f_{m+1}(x_i))\leq\epsilon/(12k)\right.\\
&\qquad\qquad \left. \text{and } \frac{1}{m}\sum_{i=1}^{m} l(f_i(x_i),h_{\bar{x},\sigma}(x_i))\leq\epsilon/12\right)
\end{aligned}$$

since $f_{m+1}\in\mathcal{F}$ and $h_{\bar{x},\sigma}\in\text{mindis}(\bar{x})$ implies that $h_{\bar{x},\sigma}$ has at most k times as many disagreements as f_{m+1}. Recalling that $k\geq 1$ and applying the triangle inequality for l, we have

$$\begin{aligned}
\mathbf{Pr}(\bar{E}\cap G) &\leq \mathbf{Pr}\left(\mathbf{er}_{f_{m+1}}(h_{\bar{x},\sigma})>\epsilon/3 \text{ and } \frac{1}{m}\sum_{i=1}^{m} l(h_{\bar{x},\sigma}(x_i),f_{m+1}(x_i))\leq\epsilon/6\right)\\
&\leq \epsilon/3
\end{aligned}$$

by Lemma 66, since $m\geq\frac{192d}{\epsilon}\ln\frac{192}{\epsilon}$. Plugging in to (7.1) yields the desired result. □

If $\langle f_i\rangle$ is a (Δ,D)-admissible sequence of functions, then $\mathbf{Pr}_{x\in D}(f_i(x)\neq f_{m+1}(x))\leq(m-i+1)\Delta$, and

$$\sum_{i=1}^{m}\mathbf{Pr}_{x\in D}(f_i(x)\neq f_{m+1}(x))\leq m(m+1)\Delta/2.$$

Thus we obtain the following corollary.

Corollary 68: *Let A be a tracking strategy that predicts using a randomly chosen hypothesis which, with probability $1-\epsilon/6$, approximately minimizes the number of disagreements on*

Theorem 67: *Let $(X, \mathcal{F})$ be a tracking problem, $d = VCdim(\mathcal{F})$, and choose $\epsilon > 0$. Suppose A is a randomized tracking algorithm which, with probability at least $1 - \epsilon/6$, predicts using an $h \in \mathcal{F}$ having at most k times the minimum number of disagreements on the previous trials. Choose a distribution D on X and*

$$m \geq \max\left(\frac{192d}{\epsilon}\ln\frac{192}{\epsilon}, \frac{72k}{\epsilon}\ln\frac{6}{\epsilon}\right).$$

Then if the sequence of targets from $\mathcal{F}$, $S = \langle f_i \rangle_{i \in \mathbf{N}}$, satisfies $\sum_{i=1}^{m} \mathbf{Pr}_{x \in D}(f_i(x) \neq f_{m+1}(x)) \leq m\epsilon/(24k)$, the probability that A makes a mistake on the $(m+1)$st trial is at most ϵ.

Proof: Fix m and k. For each $\bar{x} \in X^m$, let mindis$(\bar{x})$ be the set of all hypotheses in $\mathcal{F}$ which approximately minimize disagreements with $\mathrm{sam}_m(S, \bar{x})$ to within a factor of k.

Define F to be the event that the hypothesis chosen by A is not in mindis$(\bar{x})$.

Define F' to be the event that there are more than twice the expected number of disagreements between the previous trials and f_{m+1}, i.e.,

$$F' = \{\bar{x} \in X^m : \sum_{i=1}^{m} l(f_i(x_i), f_{m+1}(x_i)) > m\epsilon/(12k)\}.$$

Applying Lemma 65 (with $\alpha = 1$), we have

$$\mathbf{Pr}_{\bar{x} \in D^m}(F') \leq e^{-m\epsilon/(72k)} \leq \epsilon/6,$$

since $m \geq \frac{72k}{\epsilon}\ln\frac{6}{\epsilon}$.

Define $E = F \cup F'$. Then $\mathbf{Pr}(E) \leq \epsilon/3$.

For each $\bar{x} \in X^m, \sigma \in \Gamma$, let $h_{\bar{x},\sigma}$ be A's hypothesis after seeing the sequence

$$(x_1, f_1(x_1)), ..., (x_m, f_m(x_m))$$

of examples and the random sequence σ. Let

$$G = \{(\bar{x}, \sigma) \in X^m \times \Gamma : \mathbf{er}_{f_{m+1}}(h_{\bar{x},\sigma}) > \epsilon/3\},$$

be the set of sequences of points and random bits that cause A to produce an inaccurate hypothesis.

probability that there exists a hypothesis h in class $\mathcal{F}$ such that the estimate of h's error is small but the true probability that h will yield an incorrect prediction is large.

We will make use of the standard Chernoff bounds, which we state here.

Lemma 65 ([Angluin and Valiant, 1979,Littlestone, 1989b]): *Let $t \in \mathbf{N}$, and let $r_1, ..., r_t$ be independent $\{0,1\}$-valued random variables. Choose α, $0 < \alpha \leq 1$. Let $\mu = \sum_{i=1}^{t} \mathbf{Pr}(r_i = 1)$. Then*

$$\mathbf{Pr}\left(\sum_{i=1}^{t} r_i \geq (1+\alpha)\mu\right) \leq e^{-\alpha^2\mu/3}.$$

For each $h \in \mathcal{F}$, $f \in \mathcal{F}$, $m \in \mathbf{N}$, $\bar{x} \in X^m$, define

$$\mathbf{er}_f(h) = \mathbf{Pr}_{x \in D}(h(x) \neq f(x)),$$

(D is to be understood from context), and define

$$\hat{\mathbf{er}}_f(h, \bar{x}) = \frac{1}{m}\sum_{i=1}^{m} l(h(x_i), f(x_i)),$$

where $l(u,v)$ is the *discrete loss* function, i.e. $l(u,v)$ is 1 if $u = v$ and 0 otherwise. Note that $\hat{\mathbf{er}}_f$ is the empirical estimate of the error of h obtained when the (unchanging) target concept is f.

Our first lemma follows immediately from the results of [Blumer *et al.*, 1989, Theorem A3.1].

Lemma 66: *For any set X and concept class $\mathcal{F}$ over X, for any distribution D on X, for any $f \in \mathcal{F}$, for all $0 < \epsilon \leq 1/2$, if $m \geq \frac{64d}{\epsilon}\ln\frac{64}{\epsilon}$, where d is the VC-dimension of $\mathcal{F}$, then*

$$\mathbf{Pr}_{x \in D^m}(\exists h \in \mathcal{F} : \mathbf{er}_f(h) \geq \epsilon, \hat{\mathbf{er}}_f(h) < \epsilon/2) \leq \epsilon.$$

We are now ready to present the main result of this section. The following theorem shows that if a randomized tracking strategy is likely to predict with a hypothesis that approximately minimizes disagreements on the previous examples, then the probability that the algorithm makes a mistake on the next example is small.

We say that $\mathcal{F}$ is (ϵ, Δ)-trackable if there is a tracking strategy which (ϵ, Δ)-tracks $\mathcal{F}$.

To discuss issues of computational efficiency, we will need the following definitions. We say that $\mathcal{F} = \{\mathcal{F}_n : n \in \mathbf{N}\}$ is a *stratified tracking problem* if for each $n \in \mathbf{N}$, $(\mathbf{Q}^n, \mathcal{F}_n)$ is a tracking problem.[2] An algorithm for a stratified tracking problem consists of a tracking algorithm A_n for each n. We assume that the random bits are presented on an auxiliary tape, and thus accessing the next random bit in the sequence takes unit time.

We say that $A = \{A_n\}$ efficiently tracks $\mathcal{F}$ if there is a polynomial p and positive constants c and k such that for all relevant ϵ, n,

- each prediction is computed in time bounded by $p(1/\epsilon, n, b)$, where b is the number of bits needed to encode the "largest" example seen.
- at most $p(1/\epsilon, n, b)$ space is required to store information between trials,
- if $\Delta < c(\epsilon/n)^k$, A_n (ϵ, Δ)-tracks $\mathcal{F}_n$.

Note that the bound on the space required is not allowed to grow with the number of trials. Thus an efficient tracking algorithm may not, in general, keep all previously seen examples.

7.2 Increasingly unreliable evidence and hypothesis evaluation

In this section we analyze a simple tracking algorithm which ignores all examples beyond some time in the past and uses the hypothesis which disagrees with the fewest remaining examples for prediction. The results of this section, together with those of Section 7.4, show that this apparently naive algorithm is within a constant times a log factor of optimal for the classes of halfspaces and hyperrectangles. We also show that it is sufficient to only approximately minimize disagreements to within a constant factor.

As discussed in the introduction, the fraction of the considered examples disagreeing with a hypothesis can be viewed as an estimate of the probability that the hypothesis will make a mistake on the next example. In the following series of lemmas we bound the

[2] We assume rationals are encoded by encoding both the numerator and the denominator in binary.

generated.

Aldous and Vazirani [Aldous and Vazirani, 1990] studied a different version of learning in a changing environment. In their model the target concept is fixed, but the examples are generated by a Markov process rather then from a fixed distribution.

7.1 Notation and some definitions

A *tracking problem*, $(X, \mathcal{F})$ consists of a set (or *domain*) X and a family $\mathcal{F}$ of $\{0,1\}$-valued functions defined on X, called the *target class*. A $\{0,1\}$ valued function defined on X is called a *concept*. We will speak of a concept and the subset of X on which it takes value 1 interchangeably. An *example* is an element of $X \times \{0,1\}$, and a *sample* is a finite sequence of examples. A function h *agrees* (resp. *disagrees*) with an example (x, ρ) when $h(x) = \rho$ (resp. $h(x) \neq \rho$). A function is *consistent* with a sample if it agrees with all examples in the sample. We often use the discrete loss function, $l(\alpha, \beta)$, defined to be 0 when $\alpha = \beta$ and 1 otherwise, to count numbers of disagreements.

Let Γ be the set of all infinite sequences of bits, and $\mathcal{U}$ be the distribution which sets each bit in the sequence independently with probability $1/2$. A *(randomized) tracking strategy* is a mapping from $(\cup_m (X \times \{0,1\})^m) \times X \times \Gamma$ to $\{0,1\}$.

If $S = \langle f_t \rangle_{t \in \mathbf{N}}$ is a sequence of concepts and $\bar{x} \in X^n$ with $n \geq m$, the *m-sample of S generated by $\bar{x}$*, written $\mathrm{sam}_m(S, \bar{x})$, is the sequence of pairs $\langle (x_1, f_1(x_1)), ..., (x_m, f_m(x_m)) \rangle$. Informally, $\mathrm{sam}_m(S, \bar{x})$ is simply the first m examples which are used by a tracking strategy to predict $f_{m+1}(x_{m+1})$.

Let D be a probability distribution over X. If $\Delta \geq 0$, a sequence $\langle f_t \rangle_{t \in \mathbf{N}}$ of concepts is called (Δ, D)-admissible if for each $t \in \mathbf{N}$, $\mathbf{Pr}_{x \in D}(f_t(x) \neq f_{t+1}(x)) \leq \Delta$.

Let A be a tracking strategy. We say that A (ϵ, Δ)-tracks $\mathcal{F}$ if there is an $m_0 \in \mathbf{N}$ such that for all $m \geq m_0$, for all probability distributions D on X, and for all (Δ, D)-admissible sequences $S = \langle f_t \rangle_{t \in \mathbf{N}}$ of functions in $\mathcal{F}$,

$$\mathbf{Pr}_{\bar{x} \in D^{m+1}, \sigma \in \mathcal{U}}(A(\mathrm{sam}_m(S, \bar{x}), x_{m+1}, \sigma) \neq f_{m+1}(x_{m+1})) \leq \epsilon.$$

within a log factor of optimal for halfspaces and hyperrectangles. A slightly modified analysis holds for the case in which the tracking algorithm uses a hypothesis which only approximately minimizes disagreements with a suffix of the examples.

In Section 7.3, we give a general purpose algorithmic transformation turning a randomized polynomial time hypothesis finder $\mathcal{A}$ [Blumer *et al.*, 1989] which, with high probability, returns a hypothesis consistent with an input sample, into an algorithm which efficiently approximately minimizes disagreements to within a factor of $cd + 1$, where d is the VC-dimension of the target class, and c is any constant. We use a technique due to Kearns and Li [Kearns and Li, 1988], working in stages, where at each stage, we subsample according to the distribution which is uniform over the sample, hoping to get a subsample for which there is a consistent hypothesis, so that we can successfully apply $\mathcal{A}$. We then return the best hypothesis of those produced by $\mathcal{A}$ during the various stages. We use an observation of Vapnik's [Vapnik, 1982] to argue that with high probability, a hypothesis consistent with the subsample can't be too bad on the whole sample.

There is little previous work on slowly drifting concepts. Littlestone and Warmuth [Littlestone and Warmuth, 1989] describe a variant of the weighted majority algorithm where the weights are kept above some lower limit. This allows the weighted majority algorithm to recover and adapt to changes in the target. However, if the target changes k times, then their mistake bound for the weighted majority algorithm goes up by about a factor of k. It is difficult to translate these bounds into our model as our targets potentially change with each example.

In work independent with ours, Kuh, Petsche and Rivest [Kuh *et al.*, 1991] studied a variety of models in which the target drifts slowly. In their paper, they concentrated on obtaining upper bounds on the tolerable rate of drift (or, equivalently, lower bounds on the probability of a mistake, given that the target is drifting at a certain rate) for the case in which the sequence of targets is produced by an adversary which at each time has access to the earlier random examples seen by the tracking algorithm. In contrast, we assume that the sequence of targets is chosen by an adversary before any random examples are

to predict the label of the next point. To analyze such algorithms, one might imagine applying the results of Vapnik and Chervonenkis [Vapnik and Chervonenkis, 1971] to show that if for each hypothesis h in the class, we estimate the probability that h will make a mistake on the next trial by considering the fraction of the last t trials on which h made a mistake, none of these estimates will be very far from the true estimated probabilities. The movement of the target prevents us from simply applying the results of [Vapnik and Chervonenkis, 1971]. To remedy this, we first bound the probability that for any hypothesis h, the estimate we obtain is very far from the estimate we would have obtained, had the target not been moving. Then we are ready to apply uniform convergence results.

If we now apply the results of [Vapnik and Chervonenkis, 1971], however, our analysis indicates that these algorithms are more than a factor of ϵ from the best upper bounds we can prove on the maximum tolerable rate of drift. In the case of learning stationary targets, it has been observed [Vapnik, 1982] [Blumer *et al.*, 1989] that uniformly good estimates of the quality of hypotheses were not required for learning in the PAC-model [Valiant, 1984]. Instead, one only needed to bound the probability that an "ϵ-bad" hypothesis was consistent with a sequence of examples. They were then able to shave a factor of $1/\epsilon$ off the bound on the number of examples required for learning with accuracy ϵ obtained by simply applying the results of [Vapnik and Chervonenkis, 1971]. However, in our case, there may not be any hypothesis consistent with more than a few of the most recent examples. Nevertheless, given reasonable restrictions on the rate of drift there is, with high probability, some hypothesis having very few disagreements with a reasonable sized suffix of a random sequence of examples. Thus, we are able to apply another of the results of [Blumer *et al.*, 1989], which bounds the probability that any ϵ-bad hypothesis is consistent with all but a fraction $\epsilon/2$ of the examples. The number of examples required to bound this "ϵ-bad but highly consistent" probability by δ is within a constant of that for the completely consistent case. Thus, ignoring constants, the factor of $1/\epsilon$ savings is retained, reducing our tracking bounds by a factor of ϵ.

The result of this analysis is a simple "minimize disagreements" algorithm which is

precise in Section 7.1.

Many readers will notice the similarity of our model to the prediction model studied in [Haussler *et al.*, 1988] and elsewhere. The key difference is that in our model there is no single target function, but rather a succession of related target functions. Since the learner may receive only a single example before the target changes, it is unreasonable to expect that the hypotheses converge to a target. However, it is possible to bound the probability of a mistake on a trial in terms of how much the target is allowed to change between trials and the complexity of $\mathcal{F}$.

Our results include:

- a general-purpose algorithm which tolerates target movement rates up to $c_1\epsilon^2/(d \ln \frac{1}{\epsilon})$ (Theorem 67 and Corollary 69),
- a possibly more computationally efficient variant of this algorithm which tolerates target movements of up to $c_2\epsilon^2/(d^2 \ln \frac{1}{\epsilon})$ (Theorem 71), and
- bounds for the classes of halfspaces and axis-aligned hyperrectangles showing that for all n and $\epsilon < 1/12$, no algorithm can tolerate target movement greater than $c_3\epsilon^2/n$, where n is the dimension of the space from which examples are drawn (Theorem 78).[1]

In the above, the c_i's are constants, ϵ denotes the desired probability of error, and d is the VC-dimension of $\mathcal{F}$. The first general-purpose algorithm above is computationally efficient whenever the problem of finding a member of $\mathcal{F}$ which minimizes the number of disagreements with a set of examples can be solved efficiently. Its variant is computationally efficient whenever the problem of finding an element of $\mathcal{F}$ consistent with a set of examples can be solved efficiently, as is the case with both halfspaces and hyperrectangles.

Our algorithms use only the most recent t examples (rather than the entire sequence) to make their predictions. They work by either minimizing or approximately minimizing the number of disagreements with the most recent examples, and using the resulting hypothesis

[1]Since, in both the case of halfspaces and that of hyperrectangles in n-dimensional space, the first algorithm above tolerates drift rates up to a constant times $\epsilon^2/(n \ln \frac{1}{\epsilon})$, these bounds establish the fact that the first algorithm is within a constant times a log factor of optimal.

7. Tracking Drifting Concepts

In the fairy tale, Rip van Winkle slept for 20 years and when he finally woke up, he discovered that he was out of step with the world. Presumably, Rip would have been much better off if he woke up every day. However, if he woke for only one day each week or month or year, how comfortable would Rip be with the world after his 20 year slumber? This leads to the question "How long can one nap before losing touch with the world?" which is the subject of this chapter.

More formally, let D be a probability distribution on some set X and $\mathcal{F}$ be a class of $\{0,1\}$-valued functions defined on X. In the sleeper example, each $f \in \mathcal{F}$ represents a possible state of the world. When Rip van Winkle wakes for the t^{th} time, the world is in some state $f_t \in \mathcal{F}$. Rip gets x_t, a randomly drawn (w.r.t. D) element of X, and is asked for the value of $f_t(x_t)$. One interpretation is that x_t is a possible course of action, and $f_t(x_t) = 1$ when x_t is appropriate in the current world state. Just before Rip goes back to sleep, he is told the value of $f_t(x_t)$.

In other words, given $(x_1, f_1(x_1))$, $(x_2, f_2(x_2))$, ..., $(x_{t-1}, f_{t-1}(x_{t-1}))$, and a point x_t, Rip is asked to predict the value of $f_t(x_t)$. If Rip's prediction is incorrect we say that he makes a mistake on x_t. If Rip rarely makes mistakes, then he successfully tracks the state of the world. In our model, an adversary chooses the probability distribution D and the sequence of functions ahead of time, before the x_t's are generated.

The sequence of examples could be uninformative for two different reasons. First, x_1 through x_{t-1} may come from an uninteresting part of the domain. Any learning algorithm using randomly drawn examples must deal with this potential difficulty. A more severe problem is that the f_t chosen by the adversary may be unrelated to the previous f_i's. If the adversary randomly chooses f_t to be either the constant function 1 or the constant function 0, then no algorithm can expect to predict $f_t(x_t)$ correctly more than half the time. We deal with this problem with an assumption that the state of world evolves slowly. Thus the adversary must choose sequences of functions where each f_t is "close" to f_{t-1}. This is made

characterization of the learnability of classes of real-valued functions, although a detailed discussion of this belief is beyond the scope of this thesis. At the time of this writing, Alon, Ben-David, Cesa-Bianchi, and Haussler were making significant progress on this problem.

6.4 Discussion

In this chapter, we have given tight bounds on the cardinality of a subset of $\prod_i \{0, ..., r_i\}$ of a certain dimension for two generalizations of the VC-dimension: namely the pseudo dimension discussed by Pollard [Pollard, 1984] and the graph dimension introduced by Natarajan [Natarajan, 1989]. We also have used a similar technique obtain tighter bounds for another generalization of the VC-dimension introduced by Natarajan, which we have called the Natarajan dimension. The problem of obtaining tight bounds for the Natarajan dimension remains open.

In addition, we have applied this result to bound the rate of convergence of empirical estimates of the expectations of a sequence of random variables to their true expectations, obtaining bounds similar to those already derived in [Pollard, 1984] [Haussler, 1991]. These results can be extended to bound the sample size required for learning under the computational model of learnability discussed in [Haussler, 1991].

An interesting question may be asked about generalizations of bounds like those of this chapter for families Ψ of functions from $\{0, ..., n\}$ to $\{0, 1, *\}$ that are not distinguishers. In particular, we are interested in obtaining bounds on $|S|$ that grow polynomially in m. As in Theorem 47, we can see that such bounds are impossible if Ψ is not a distinguisher, since if Ψ fails to distinguish $a_1, a_2 \in \{0, ..., r\}$, clearly the set $S = \{a_1, a_2\}^m$ has Ψ-dimension 0, yet has 2^m elements. In the simplest case, $n = 2$ and $\Psi = \{\psi\}$, where

$$\psi(a) = \begin{cases} a & \text{if } a \neq 2 \\ * & \text{otherwise;} \end{cases}$$

so that Ψ fails to distinguish $(0, 2)$ and $(1, 2)$. However, if we say $\vec{x}, \vec{y} \in \{0, ..., r\}^m$ are Ψ-separated if there exists i such that Ψ distinguishes x_i and y_i, we may ask: What is the largest subset of $\{0, ..., r\}^m$ of Ψ-dimension at most d such that its elements are pairwise Ψ-separated? By the above results, it is already known that for those Ψ which are distinguishers, the size of the largest such set grows polynomially in m. We conjecture that this holds for all Ψ. We believe that the proof of this would result in a pleasant

Taking logs and rearranging terms yields the following equivalent expression:

$$\frac{\alpha m}{2k} \geq d\left(\ln m + \ln \frac{ke}{d}\right) + \ln 4/\delta. \tag{6.2}$$

Since by the preceding lemma for any $\lambda \in \mathbf{R}, 0 < \lambda < 1$,

$$\ln m \leq \left(\frac{\lambda\alpha}{kd}\right) m + \left(\ln \frac{kd}{\lambda\alpha e}\right),$$

the following is sufficient to guarantee Inequality (6.2):

$$\begin{aligned} \frac{\alpha m}{2k} &\geq d\left(\frac{\lambda\alpha}{kd} m + \ln \frac{kd}{\lambda\alpha e} + \ln \frac{ke}{d}\right) + \ln 4/\delta \\ &= \frac{\lambda\alpha}{k} m + d \ln \frac{k^2}{\lambda\alpha} + \ln 4/\delta. \end{aligned}$$

Solving for m yields

$$m \geq \frac{2k}{\alpha(1-2\lambda)}\left(2d \ln \frac{k}{\sqrt{\lambda\alpha}} + \ln \frac{4}{\delta}\right)$$

and resubstituting $k = \frac{4}{\alpha\nu}$ gives

$$m \geq \frac{8}{\alpha^2\nu(1-2\lambda)}\left(2d \ln \frac{4}{\alpha\nu\sqrt{\lambda\alpha}} + \ln \frac{4}{\delta}\right).$$

We choose $\lambda = 1/18$ for readability, yielding

$$m \geq \frac{9}{\alpha^2\nu}\left(2d \ln \frac{17}{(\alpha\sqrt{\alpha})\nu} + \ln \frac{4}{\delta}\right)$$

which is the desired bound. □

For comparison, we give the following theorem from [Haussler, 1991], which was obtained using a completely different technique, due to Pollard [Pollard, 1990].

Theorem 64: *Let F be a set of random variables on Ω taking values in $[0,1]$. Assume $0 < \nu \leq 4/d$, $0 < \alpha < 1$ and $m \geq 1$. Suppose that $\vec{\xi}$ is generated by m independent random draws according to the fixed measure D on Ω. Suppose also that P-dim$(F) \leq d$. Then*

$$\mathbf{Pr}\left\{\exists f \in F : d_\nu(\hat{E}_{\vec{\xi}}(f), E(f)) > \alpha\right\} \leq 8\left(\frac{16e}{\alpha\nu} \ln \frac{16e}{\alpha\nu}\right)^d e^{-\alpha^2\nu m/8}.$$

Moreover, for

$$m \geq \frac{8}{\alpha^2\nu}\left(2d \ln \frac{8e}{\alpha\nu} + \ln \frac{8}{\delta}\right),$$

this probability is less than δ.

Theorem 63: *Let F be a set of random variables on Ω taking values in $[0,1]$. Assume $\nu > 0$, $0 < \alpha < 1$ and $m \geq 1$. Suppose that $\vec{\xi}$ is generated by m independent random draws according to the fixed measure D on Ω. Suppose also that P-dim$(F) \leq d$. Then*

$$\mathbf{Pr}\left\{\exists f \in F : d_\nu(\hat{E}_{\vec{\xi}}(f), E(f)) > \alpha\right\} \leq 4\left(\frac{4}{\alpha\nu}\right)^d \left(\frac{em}{d}\right)^d e^{-\alpha^2\nu m/8}.$$

Moreover, for

$$m \geq \frac{9}{\alpha^2\nu}\left(2d\ln\frac{17}{(\alpha\sqrt{\alpha})\nu} + \ln\frac{4}{\delta}\right),$$

this probability is less than δ.

Proof: First, from Corollary 61, we have that

$$\mathcal{N}(\alpha\nu/8, F_{|\vec{\xi}}, d_{L^1}) \leq \sum_{i=0}^{d}\binom{m}{i}\left\lfloor\frac{4}{\alpha\nu}\right\rfloor^i.$$

Using the well known combinatorial identity that

$$\sum_{i=0}^{d}\binom{m}{i} \leq (em/d)^d$$

and substituting

$$\left(\frac{4}{\alpha\nu}\right)^d$$

for each

$$\left\lfloor\frac{4}{\alpha\nu}\right\rfloor^i,$$

we get

$$\mathcal{N}(\alpha\nu/8, F_{|\vec{\xi}}, d_{L^1}) \leq \left(\frac{4}{\alpha\nu}\right)^d\left(\frac{em}{d}\right)^d.$$

Applying Theorem 59 yields the first result.

Now, we wish to determine a lower bound on m which guarantees that

$$4\left(\frac{4}{\alpha\nu}\right)^d\left(\frac{em}{d}\right)^d e^{-\alpha^2\nu m/8} \leq \delta.$$

Set $k = \frac{4}{\alpha\nu}$. Then the above expression simplifies to

$$4k^d\left(\frac{em}{d}\right)^d e^{-\frac{\alpha m}{2k}} \leq \delta.$$

Corollary 61: *Let* $m \in \mathbf{Z}^+$. *Let* $F \subseteq [0,1]^m$ *be such that P-dim*$(F) \leq d$. *Let* $\epsilon \in \mathbf{R}^+$. *Then*

$$\mathcal{N}(\epsilon, F, d_{L^1}) \leq \sum_{i=0}^{d} \binom{m}{i} \left(\left\lfloor \frac{1}{2\epsilon} \right\rfloor \right)^i$$

Proof: As discussed above

$$\mathcal{N}(\epsilon, F, d_{L^1}) \leq \mathcal{N}(\epsilon, F, d_{L^\infty}).$$

The corollary then follows from the previous lemma. □

The technique by which we obtain bounds on the sample size necessary for the uniform convergence of estimates to true means for a sequence of random variables has elements which are similar to that used to in [Anthony *et al.*, 1990] improve the bounds of [Blumer *et al.*, 1989]. The following approximation is useful in this derivation.

Lemma 62 ([Anthony *et al.*, 1990]): *Let* $x, y \in \mathbf{R}^+$. *Then*

$$\ln x \leq xy - \ln ey.$$

Proof: Fix $y \in \mathbf{R}^+$. Consider $f : \mathbf{R}^+ \to \mathbf{R}$ defined by

$$f(x) = xy - \ln exy.$$

Then

$$f'(x) = y - 1/x.$$

Clearly, $f'(x)$ is positive when $x > 1/y$ and negative when $x < 1/y$ and f is continuous and differentiable over its domain, so f assumes its minimum at $1/y$ and

$$f(1/y) = y(1/y) - \ln ey(1/y) = 0.$$

So $f(x) \geq 0$ for all $x \in \mathbf{R}^+$, which yields the desired result. □

Finally, we are ready to bound the sample size necessary to ensure that with high probability an empirical estimate of the expected value of a random variable chosen from a set of a small P-dimension is accurate.

Next, we wish to show that P-dim$(T) \leq d$. Let $\vec{\imath} = (i_1, ..., i_k)$ be shattered by T and let

$$(\psi_{P,y_1}, ..., \psi_{P,y_k})$$

witness this shattering. We claim that $2\epsilon\vec{y}$ witnesses F's shattering of $\{i_1, ..., i_k\}$. Choose $\vec{b} \in \{0,1\}^k$. Let $\vec{t} \in T$ satisfy $\vec{b}$. Choose $\vec{f} \in F$, such that $\beta(\vec{f}) = \vec{t}$.

If $b_j = 1$, we have $t_{i_j} \geq y_j$ which is equivalent to

$$\left\lfloor \frac{f_{i_j}}{2\epsilon} \right\rfloor \geq y_j$$

which implies

$$\frac{f_{i_j}}{2\epsilon} \geq y_j$$

since $x \geq \lfloor x \rfloor$ for all $x \in \mathbf{R}$. Finally, the previous inequality implies

$$f_{i_j} \geq 2\epsilon y_j.$$

So if $b_j = 1$, $f_{i_j} \geq 2\epsilon y_j$.

Suppose $b_j = 0$ and $f_{i_j} \geq 2\epsilon y_j$. This implies $f_{i_j}/2\epsilon \geq y_j$, which in turn implies

$$t_{i_j} = \left\lfloor \frac{f_{i_j}}{2\epsilon} \right\rfloor \geq y_j,$$

since $y_j \in \mathbf{Z}$. But this is a contradiction, since $t_{i_j} < y_j$, which holds because $b_j = 0$ and $\vec{t}$ satisfies $\vec{b}$. So if $b_j = 0$, we have $f_{i_j} < 2\epsilon y_j$.

In the preceding two paragraphs we have established that for all $j, 1 \leq j \leq k$, we have $f_{i_j} \geq 2\epsilon y_j$ if and only if $b_j = 1$, and thereby that $\vec{f}$ satisfies $\vec{b}$. Since $\vec{b}$ was chosen arbitrarily, $\{i_1, ..., i_k\}$ is shattered by F. Since $(i_1, ..., i_k)$ was chosen arbitrarily, P-dim$(T) \leq$ P-dim$(F) = d$.

Now, by Corollary 50,

$$|T| \leq \sum_{i=0}^{d} \binom{m}{i} \left\lfloor \frac{1}{2\epsilon} \right\rfloor^i.$$

Since H is an ϵ-cover of F and $|T| = |H|$, we have

$$\mathcal{N}(\epsilon, F, d_{L^\infty}) \leq \sum_{i=0}^{d} \binom{m}{i} \left\lfloor \frac{1}{2\epsilon} \right\rfloor^i,$$

which completes the proof. □

we have

$$\{0,1\}^k \subseteq \vec{\psi}(F_{|_I}).$$

Here we say that $\vec{y}$ (rather than $\vec{\psi}$, to save notation) witnesses F's RP-shattering of I and that $f \in F$ satisfies $\vec{b} \in \{0,1\}^k$ if and only if $\vec{\psi}(f_{|_I}) = \vec{b}$. The RP-dimension of F is the cardinality of the largest subset of X shattered by F.

Note that an element of $[0,1]^m$ may be viewed as a function from $[m]$ to $[0,1]$, so we may naturally intepret the definition of RP-dimension as applying to subsets of $[0,1]^m$ as well.

Now we wish to show that if a subset of a product of closed intervals of $\mathbf{R}$ has small RP-dimension, then it has a small ϵ-cover in the d_{L^∞} metric.

Lemma 60: *Let $m \in \mathbf{Z}^+$. Let $F \subseteq [0,1]^m$ have RP-dimension at most d. Let $\epsilon \in \mathbf{R}^+$. Then*

$$\mathcal{N}(\epsilon, F, d_{L^\infty}) \leq \sum_{i=0}^{d} \binom{m}{i} \left\lfloor \frac{1}{2\epsilon} \right\rfloor^i$$

Proof: Define $\beta : [0,1]^m \rightarrow \{0, ..., \left\lfloor \frac{1}{2\epsilon} \right\rfloor\}^m$ by $\beta(\vec{s}) = \vec{t}$, where $t_i = \left\lfloor \frac{f_i}{2\epsilon} \right\rfloor$ for all $i, 1 \leq i \leq m$. Let $T = \beta(F)$. Let

$$H = \{2\epsilon\vec{t} + (\epsilon, \epsilon, ..., \epsilon) : \vec{t} \in T\}.$$

First, we claim that H is an ϵ-cover for F with respect to the d_{L^∞} metric. Choose $\vec{f} \in F$. Let $\vec{h} = 2\epsilon\beta(\vec{f}) + (\epsilon, \epsilon, ..., \epsilon)$. Choose $i, 1 \leq i \leq m$. Then we have

$$\begin{aligned} |f_i - h_i| &= \left| f_i - \left(2\epsilon \left\lfloor \frac{f_i}{2\epsilon} \right\rfloor + \epsilon \right) \right| \\ &= 2\epsilon \left| \frac{f_i}{2\epsilon} - \left\lfloor \frac{f_i}{2\epsilon} \right\rfloor - \frac{1}{2} \right| \\ &\leq \epsilon. \end{aligned}$$

Since i was chosen arbitrarily,

$$d_{L^\infty}(f,h) = \max\{|f_i - h_i| : 1 \leq i \leq m\} \leq \epsilon.$$

Since $\vec{f} \in F$ was chosen arbitrarily, H is an ϵ-cover for F.

Denote by $\mathcal{N}(\epsilon, F_{|_{\vec{\xi}}}, d_{L^\infty})$ the size of the smallest ϵ-cover of $F_{|_{\vec{\xi}}}$ in the d_{L^∞} metric by elements of $\mathbf{R}^m$. Since clearly for all $\vec{x}, \vec{y} \in \mathbf{R}^m$, $d_{L^1}(\vec{x}, \vec{y}) \leq d_{L^\infty}(\vec{x}, \vec{y})$, any ϵ-cover in the d_{L^∞} metric also serves as a ϵ-cover in the d_{L^1} metric, which implies

$$\mathcal{N}(\epsilon, F_{|_{\vec{\xi}}}, d_{L^1}) \leq \mathcal{N}(\epsilon, F_{|_{\vec{\xi}}}, d_{L^\infty}).$$

We are now ready for the following theorem. Similar results are given in [Dudley, 1984] [Pollard, 1984] [Vapnik, 1982]. In general, these theorems bound deviation of estimates $\hat{E}_{\vec{\xi}}(f)$ from true means $E(f)$ for functions f in F in terms of sizes of ϵ-covers for $F_{|_{\vec{\xi}}}$.

Theorem 59 ([Haussler, 1991]): *Let F be a set of random variables on Ω taking values in $[0,1]$. Assume $\nu > 0$, $0 < \alpha < 1$ and $m \geq 1$. Suppose that $\vec{\xi} \in \Omega^m$ is generated by m independent random draws according to the fixed measure D on Ω. Let*

$$p(\alpha, \nu, m) = \mathbf{Pr}\left\{\exists f \in F : d_\nu(\hat{E}_{\vec{\xi}}(f), E(f)) > \alpha\right\}.$$

Then

$$p(\alpha, \nu, m) \leq 2E\left(\min(2\mathcal{N}(\alpha\nu/8, F_{|_{\vec{\xi}}}, d_{L^1})e^{-\alpha^2\nu m/8}, 1)\right).$$

Let us generalize the definition of the P-dimension given earlier for sets of integer vectors to sets of real valued functions, and let us call the resulting notion that RP-dimension. Let $RP = \{\psi_{RP,y} : y \in \mathbf{R}\}$, where, again

$$\psi_{RP,y}(x) = \begin{cases} 1 & \text{if } x \geq y \\ 0 & \text{otherwise.} \end{cases}$$

Let F be a set of real valued functions defined on some linearly ordered domain X. Let $I = \{x_1, ..., x_k\} \subseteq X$, with $x_1 < x_2 < \cdots < x_k$. For $f \in F$, let

$$f_{|_I} = (f(x_1), ..., f(x_k)).$$

Define

$$F_{|_I} = \{f_{|_I} : f \in F\}.$$

We say that I is RP-shattered by F if there exists $\vec{y} \in \mathbf{R}^k$ such that if

$$\vec{\psi} = (\psi_{RP,y_1}, ..., \psi_{RP,y_k}),$$

ϵ-cover for each $\epsilon \in \mathbf{R}^+$. In this case, we let $\mathcal{N}(\epsilon, T, d)$ denote the cardinality of the smallest ϵ-cover of T (w.r.t. S and d).

Now, we define the metric relative to which we prove uniform convergence results in this section. This metric was introduced and its utility as a measure of accuracy for an approximation of a function was discussed in [Haussler, 1991]. For each $\nu \in \mathbf{R}^+$, define $d_\nu : \mathbf{R}^+ \times \mathbf{R}^+ \to \mathbf{R}^+$ by

$$d_\nu(r, s) = \frac{|r - s|}{\nu + r + s}.$$

It is straightforward but tedious to verify that for all $\nu \in \mathbf{R}^+$, d_ν is a metric on $\mathbf{R}^+$.

Let $(\Omega, \mathcal{B}, D)$ be probability space with D a probability measure on the set Ω, and $\mathcal{B}$ some appropriate σ-algebra on Ω. Let F be a set of random variables on Ω. For $m \geq 1$, denote by Ω^m the m-fold product space with the usual product probability measure. For any

$$\vec{\xi} = (\xi_1, ..., \xi_m) \in \Omega^m$$

and $f \in F$, let

$$\hat{E}_{\vec{\xi}}(f) = \frac{1}{m} \sum_{i=1}^{m} f(\xi_i).$$

and

$$F_{|_{\vec{\xi}}} = \{(f(\xi_1), ..., f(\xi_m)) : f \in F\}.$$

We can view $F_{|_{\vec{\xi}}}$ as a subspace of the metric space $(\mathbf{R}^m, d_{L^1})$, where d_{L^1} is the usual L^1 metric, i.e., for any $\vec{x} = (x_1, ..., x_m)$ and $\vec{y} = (y_1, ..., y_m)$ in $\mathbf{R}^m$,

$$d_{L^1}(\vec{x}, \vec{y}) = \frac{1}{m} \sum_{i=1}^{m} |x_i - y_i|.$$

Also, we denote by $\mathcal{N}(\epsilon, F_{|_{\vec{\xi}}}, d_{L^1})$ the size of the smallest ϵ-cover of $F_{|_{\vec{\xi}}}$ in the d_{L^1} metric by elements of $\mathbf{R}^m$.

Similary, we can view $F_{|_{\vec{\xi}}}$ as a subspace of $(\mathbf{R}^m, d_{L^\infty})$, where d_{L^∞} is defined as follows. For $\vec{x} = (x_1, ..., x_m)$ and $\vec{y} = (y_1, ..., y_m)$ in $\mathbf{R}^m$,

$$d_{L^\infty}(\vec{x}, \vec{y}) = \max\{|x_i - y_i| : 1 \leq i \leq m\}.$$

$$
\begin{aligned}
|S| &\leq \left[\sum_{i=0}^{d}\sum_{S\in\Gamma_{(m-1),i}}\prod_{k\in S}\binom{r_k+1}{2}\right]+\binom{r_m+1}{2}\sum_{i=0}^{d-1}\sum_{S\in\Gamma_{(m-1),i}}\prod_{k\in S}\binom{r_k+1}{2}\\
&= \left[\sum_{i=0}^{d}\sum_{S\in\Gamma_{(m-1),i}}\prod_{k\in S}\binom{r_k+1}{2}\right]+\sum_{i=0}^{d-1}\sum_{S\in\Gamma_{(m-1),i}}\prod_{k\in S\cup\{m\}}\binom{r_k+1}{2}\\
&= \left[1+\sum_{i=1}^{d}\sum_{S\in\Gamma_{(m-1),i}}\prod_{k\in S}\binom{r_k+1}{2}\right]+\sum_{i=1}^{d}\sum_{S\in\Gamma_{(m-1),(i-1)}}\prod_{k\in S\cup\{m\}}\binom{r_k+1}{2}\\
&= 1+\sum_{i=1}^{d}\left\{\left[\sum_{S\in\Gamma_{(m-1),i}}\prod_{k\in S}\binom{r_k+1}{2}\right]+\left[\sum_{S\in\Gamma_{(m-1),(i-1)}}\prod_{k\in S\cup\{m\}}\binom{r_k+1}{2}\right]\right\}\\
&= 1+\sum_{i=1}^{d}\left\{\left[\sum_{S\in\Gamma_{m,i},m\notin S}\prod_{k\in S}\binom{r_k+1}{2}\right]+\left[\sum_{S\in\Gamma_{m,i},m\in S}\prod_{k\in S}\binom{r_k+1}{2}\right]\right\}\\
&= \sum_{i=0}^{d}\sum_{S\in\Gamma_{m,i}}\prod_{k\in S}\binom{r_k+1}{2}
\end{aligned}
$$

which completes the induction. □

Theorem 51 can now easily be established.

6.3 An application

In this section, we give an application of Corollary 50, bounding the sample size necessary to obtain uniformly good empirical estimates for the expectations of all random variables of a given class S in terms of a generalization of the definition of P-dimension given above to classes of real valued functions, in this case, random variables. We will measure the deviation of the estimates from the true expectations using a metric introduced in [Haussler, 1991]. These results can be extended to bound the sample size necessary for learning according to the computational model of learning discussed in [Haussler, 1991], an extension of that introduced in [Valiant, 1984] which incorporates additional methods from previous work in Pattern Recognition.

We begin with some definitions. Let (T, d) be a bounded metric space (see Appendix A for a definition). For any $\epsilon \in \mathbf{R}^+$, a finite set N is an ϵ-cover for T if and only if for all $x \in T$, there exists $y \in N$ with $d(x, y) \leq \epsilon$. We say T is totally bounded if T has a finite

Since each of the above sets are disjoint and their union is all of S, we have

$$|S| = |S_-| + \sum_{u=0}^{r_m-1} \sum_{v=u+1}^{r_m} |S_{uv}|.$$

Using the same argument as in the previous lemma, under the inductive hypothesis that the lemma holds for all sets S of vectors of $m-1$ elements, we have

$$|S_-| \leq \sum_{i=0}^{d} \sum_{S \in \Gamma_{(m-1),i}} \prod_{k \in S} \binom{r_k + 1}{2}.$$

Now, we wish to establish the following claim under the same inductive hypothesis.

Claim 58: *For all $u, v \in \mathbf{N}, 0 \leq u < v \leq r_m$, we have*

$$|S_{uv}| \leq \sum_{i=0}^{d-1} \sum_{S \in \Gamma_{(m-1),i}} \prod_{k \in S} \binom{r_k + 1}{2}.$$

Proof (of Claim): Choose $u, v \in \mathbf{N}, 0 \leq u < v \leq r_m$. We will show that the N-dimension of S_{uv} is at most $d-1$. The claim then follows by an argument similar to that of Claim 55. Let $\vec{\imath} = (i_1, ..., i_l)$ be a sequence of indices shattered by S_{uv}.

Now we show that $(i_1, ..., i_l, m)$ is shattered by S. Let $(\psi_{N,y_1,z_1}, ..., \psi_{N,y_l,z_l})$ be witness of S_{uv}'s N-shattering of $\vec{\imath}$.

We claim that $(\psi_{N,y_1,z_1}, ..., \psi_{N,y_l,z_l}, \psi_{N,u,v})$ witnesses S's N-shattering of $(i_1, ..., i_l, m)$. Choose $\vec{b} \in \{0,1\}^{l+1}$. Let $\vec{s} \in S_{uv}$ satisfy $(b_1, ..., b_l)$ (with respect to $\vec{\imath}$).

If $b_{l+1} = 1$, then $\vec{s}$ satisfies $\vec{b}$, and if $b_{l+1} = 0$, then

$$(s_1, ..., s_{m-1}, \alpha(s_1, ..., s_{m-1})) = (s_1, ..., s_{m-1}, u)$$

satisfies $\vec{b}$. Since $\vec{b}$ was chosen arbitrarily, $(i_1, ..., i_l, m)$ is N-shattered by S. Since by assumption the N-dimension of S is no greater than d, we have $l \leq d-1$. Since $\vec{\imath}$ was chosen arbitrarily, the N-dimension of S_{uv} is no greater than $d-1$, which completes our proof of this claim, by the discussion above. □

From the previous two claims, we have that

This completes the induction. □

Theorem 49 easily follows from the previous lemmas together with the discussion relating $GP_{\max}$ to $G_{\max}$ and $P_{\max}$.

Next, we turn to Theorem 51. The lower bound was established in Lemma 53. We obtain the upper bound with the following lemma, the proof of which is similar to that of Lemma 54.

Lemma 57: *Let* $d, m \in \mathbf{Z}^+, r_1, ..., r_m \in \mathbf{N}$ *be such that* $d \leq m$. *Let*

$$S \subseteq X = \prod_{i=1}^{m} \{0, ..., r_i\}$$

be such that N-dim$(S) \leq d$. *Then*

$$|S| \leq \sum_{i=0}^{d} \sum_{S \in \Gamma_{m,i}} \prod_{k \in S} \binom{r_k + 1}{2}.$$

Proof: As before, our proof is by double induction on m and d.

Using the same argument as the previous lemma, we can establish this lemma for the case $d = 0$.

Next, suppose that $d = m$. By partitioning the elements of the domain as discussed above, we can see that

$$\begin{aligned} |X| &\leq \sum_{i=0}^{m} \sum_{S \in \Gamma_{m,i}} \prod_{k \in S} r_k \\ &\leq \sum_{i=0}^{m} \sum_{S \in \Gamma_{m,i}} \prod_{k \in S} \binom{r_k + 1}{2} \end{aligned}$$

so since $S \subseteq X$, certainly

$$|S| \leq \sum_{i=0}^{m} \sum_{S \in \Gamma_{m,i}} \prod_{k \in S} \binom{r_k + 1}{2}.$$

Now, choose $d, m \in \mathbf{Z}^+$ such that $0 < d < m$. Define α and S_- as in the previous lemma and for each pair of distinct elements $u, v \in \mathbf{N}, 0 \leq u < v \leq r_m$, define

$$S_{uv} = \{\vec{s} \in S - S_- : s_m = v, \alpha(s_1, ..., s_{m-1}) = u\}.$$

Claim 56: *For all* $n \in \mathbf{N}, 1 \leq n \leq r_m$,

$$|S_n| \leq \sum_{i=0}^{d-1} \sum_{S \in \Gamma_{(m-1),i}} \prod_{k \in S} r_k.$$

Proof (of Claim 56): Choose $n \in \{1, ..., r_m\}$. We will show that the GP-dimension of S_n is at most $d-1$. The claim then follows by an argument similar to that of the previous claim. Let $\vec{\imath} = (i_1, ..., i_l)$ be a sequence of indices GP-shattered by S_n. Note that $m \notin \{i_1, ..., i_l\}$, since $s_m = n$ for all $\vec{s} \in S_n$.

Now we show that $(i_1, ..., i_l, m)$ is GP-shattered by S. Let $(\psi_{GP,y_1}, ..., \psi_{GP,y_l})$ be the witness of S_n's GP-shattering of $\vec{\imath}$. Consider $(\psi_{GP,y_1}, ..., \psi_{GP,y_l}, \psi_{GP,n})$. Choose $\vec{b} \in \{0,1\}^{l+1}$. Let $\vec{s} \in S_n$ satisfy $(b_1, ..., b_l)$ (with respect to $\vec{\imath}$).

If $b_{l+1} = 1$, then $\vec{s}$ satisfies $\vec{b}$, and if $b_{l+1} = 0$, then

$$(s_1, ..., s_{m-1}, \alpha(s_1, ..., s_{m-1}))$$

satisfies $\vec{b}$. Since $\vec{b}$ was chosen arbitrarily, $(i_1, ..., i_l, m)$ is GP-shattered by S. Since by assumption the GP-dimension of S is no greater than d, we have $l \leq d-1$. Since $\vec{\imath}$ was chosen arbitrarily, the GP-dimension of S_n is no greater than $d-1$, which is sufficient to prove this claim, as discussed above. □

From the previous two claims, we have that

$$\begin{aligned}
|S| &\leq \left[\sum_{i=0}^{d} \sum_{S \in \Gamma_{(m-1),i}} \prod_{k \in S} r_k\right] + r_m \sum_{i=0}^{d-1} \sum_{S \in \Gamma_{(m-1),i}} \prod_{k \in S} r_k \\
&= \left[\sum_{i=0}^{d} \sum_{S \in \Gamma_{(m-1),i}} \prod_{k \in S} r_k\right] + \sum_{i=0}^{d-1} \sum_{S \in \Gamma_{(m-1),i}} \prod_{k \in S \cup \{m\}} r_k \\
&= \left[1 + \sum_{i=1}^{d} \sum_{S \in \Gamma_{(m-1),i}} \prod_{k \in S} r_k\right] + \sum_{i=1}^{d} \sum_{S \in \Gamma_{(m-1),(i-1)}} \prod_{k \in S \cup \{m\}} r_k \\
&= 1 + \sum_{i=1}^{d} \left\{ \left[\sum_{S \in \Gamma_{(m-1),i}} \prod_{k \in S} r_k\right] + \left[\sum_{S \in \Gamma_{(m-1),(i-1)}} \prod_{k \in S \cup \{m\}} r_k\right] \right\} \\
&= 1 + \sum_{i=1}^{d} \left\{ \left[\sum_{S \in \Gamma_{m,i}, m \notin S} \prod_{k \in S} r_k\right] + \left[\sum_{S \in \Gamma_{m,i}, m \in S} \prod_{k \in S} r_k\right] \right\} \\
&= \sum_{i=0}^{d} \sum_{S \in \Gamma_{m,i}} \prod_{k \in S} r_k.
\end{aligned}$$

establishing the result in this case.

Now, choose $d, m \in \mathbf{Z}^+$ such that $0 < d < m$. Define $\pi : X \to \prod_{i=1}^{m-1}\{0, ..., r_i\}$ by

$$\pi(\vec{s}) = (s_1, ..., s_{m-1}).$$

Define

$$\alpha : \pi(S) \to \{0, ..., r_m\}$$

by

$$\alpha(w_1, ..., w_{m-1}) = \min\{v : (w_1, ..., w_{m-1}, v) \in S\}$$

Define

$$S_- = \{(s_1, ..., s_{m-1}, \alpha(s_1, ..., s_{m-1})) : \vec{s} \in S\}$$

and for each $n \in \mathbf{N}, 1 \leq n \leq r_m$, define

$$S_n = \{\vec{s} \in S - S_- : s_m = n\}.$$

Since the above sets are disjoint and their union is all of S, we have

$$|S| = |S_-| + \sum_{n=1}^{r_m} |S_n|.$$

Let us make the inductive assumption that the bound (6.1) holds for all sets S of vectors of $m-1$ elements. We claim that this implies the following.

Claim 55:

$$|S_-| \leq \sum_{i=0}^{d} \sum_{S \in \Gamma_{(m-1),i}} \prod_{k \in S} r_k.$$

Proof (of Claim 55): The restriction of π to S_- is 1-1 by construction of S_-. The set $\pi(S_-)$ has GP-dimension no greater than d since any set of indices shattered by $\pi(S_-)$ is also shattered by S_-, and therefore by S. By the induction hypothesis,

$$|\pi(S_-)| \leq \sum_{i=0}^{d} \sum_{S \in \Gamma_{(m-1),i}} \prod_{k \in S} r_k,$$

so since π's restriction to S_- is 1-1, the claim is verified. □

Next, under the same induction hypothesis, we make the following claim.

We can see that the G-, P-, GP- and N-dimensions of S are all no less than d, since for each of the definitions of shattering, any sequence consisting of d distinct elements of $[m]$ is shattered, since it is trivially N-shattered (taking $\vec{y} = (0, 0, ..., 0)$, $\vec{z} = (1, 1, ..., 1)$, for instance), and as discussed previously, the N-shattering of a sequence implies its G-, P- and GP-shattering.

We can see that S's cardinality is as given in the lemma by breaking the elements of S up into subsets consisting of the elements with exactly i non-zero elements, $0 \leq i \leq d$, and for each i further breaking these up according to which i elements are nonzero. □

For our next lemma, we give an upper bound on the cardinality of sets of a given GP-dimension, and thereby that of sets of a given G- or P-dimension. Our argument is a generalization of that given by Sauer in [Sauer, 1972], and is similar to Natarajan's generalization of this argument in [Natarajan, 1989].

Lemma 54: *Let* $d, m \in \mathbf{Z}^+, r_1, ..., r_m \in \mathbf{N}$ *be such that* $d \leq m$. *Let*

$$S \subseteq X = \prod_{i=1}^{m} \{0, ..., r_i\}$$

be such that GP-dim$(S) \leq d$. *Then*

$$|S| \leq \sum_{i=0}^{d} \sum_{S \in \Gamma_{m,i}} \prod_{k \in S} r_k. \tag{6.1}$$

Proof: Our proof is by double induction on m and d.

First we consider the case in which $d = 0$. Here, the bound (6.1) reduces to $|S| \leq 1$. If $|S| > 1$, then S must have two distinct elements $\vec{s}$ and $\vec{t}$. Let i be an index on whose entry $\vec{s}$ and $\vec{t}$ differ. Then $\{i\}$ is shattered by S, so the GP-dimension of S is at least 1, which contradicts the assumption that $d = 0$, so $|S| \leq 1$ and the lemma holds.

Next, suppose that $d = m$. By partitioning the elements of the domain as discussed above, we can see that

$$|X| \leq \sum_{i=0}^{m} \sum_{S \in \Gamma_{m,i}} \prod_{k \in S} r_k.$$

so since $S \subseteq X$, certainly

$$|S| \leq \sum_{i=0}^{m} \sum_{S \in \Gamma_{m,i}} \prod_{k \in S} r_k,$$

6.2 Proofs of the results

We begin by exhibiting large sets of a given G-, P-, GP-, and N-dimension.

Lemma 53: *Let $d, m \in \mathbf{Z}^+, r_1, ..., r_m \in \mathbf{N}$ be such that $d \leq m$. Then there exists*

$$S \subseteq X = \prod_{i=1}^{m} \{0, ..., r_i\}$$

such that S has G-, P-, GP-, and N-dimension d and

$$|S| = \sum_{i=0}^{d} \sum_{S \in \Gamma_{m,i}} \prod_{k \in S} r_k.$$

Proof: Define S to be all the elements of X with at most d nonzero entries. We claim S has G-, P-, GP- and N-dimension d, and $|S|$ is as given above.

To prove that the G-, P-, GP- and N-dimensions of S are all no greater than d, it is sufficient to prove that G-dim$(S) \leq d$ and P-dim$(S) \leq d$, since as discussed above

$$\begin{aligned} \text{N-dim}(S) &\leq \text{GP-dim}(S) \\ \text{GP-dim}(S) &\leq \text{P-dim}(S) \\ \text{GP-dim}(S) &\leq \text{G-dim}(S). \end{aligned}$$

First, we show that G-dim$(S) \leq d$. Assume G-dim$(S) > d$ for contradiction. Let $(\psi_{G,y_1}, ..., \psi_{G,y_k})$ witness S's G-shattering of $(i_1, ..., i_k)$, where $k > d$. Form $\vec{b} \in \{0,1\}^k$ by

$$b_i = \begin{cases} 0 & \text{if } y_i = 0 \\ 1 & \text{otherwise} \end{cases}.$$

Let $\vec{s} \in S$ satisfy $\vec{b}$. Let $\vec{t} = \vec{s}_{|\vec{r}}$. By definition of G-shattering, we have $t_i \neq y_i$ if $y_i = 0$ and $t_i = y_i$ if $y_i \neq 0$, so $t_i \neq 0$ for all i, which implies $s_{i_j} \neq 0$ for all $j \leq k$ which contradicts the definition of S, since $k > d$.

Next, we need to show that P-dim$(S) \leq d$. Again, assume P-dim$(S) > d$ for contradiction. Let $(\psi_{P,y_1}, ..., \psi_{P,y_k})$ witness S's P-shattering of $(i_1, ..., i_k)$, where again $k > d$. Let $\vec{s} \in S$ satisfy $(0, 0, ..., 0)$. Since $y_j > s_{i_j}$ for all $j, 1 \leq j \leq k$, we have $y_j > 0$ for all $j, 1 \leq j \leq k$. Let $\vec{t} \in S$ satisfy $(1, 1, ..., 1)$. Since $t_{i_j} \geq y_j$ for all $j, 1 \leq j \leq k$, we have $t_{i_j} > 0$ for all $j, 1 \leq j \leq k$, which again contradicts the definition of S.

Theorem 49: *For all* $d, m \in \mathbf{Z}^+, r_1, ..., r_m \in \mathbf{N}$ *such that* $d \leq m$,

$$\begin{aligned} G_{\max}(d, m, r_1, ..., r_m) &= \\ P_{\max}(d, m, r_1, ..., r_m) &= \\ GP_{\max}(d, m, r_1, ..., r_m) &= \sum_{i=0}^{d} \sum_{S \in \Gamma_{m,i}} \prod_{k \in S} r_k. \end{aligned}$$

When there is an $r \in \mathbf{N}$ such that $r_i = r$ for all $i, 1 \leq i \leq m$, we obtain the following corollary, which is useful for obtaining learning results such as those in [Haussler, 1991].

Corollary 50: *Let* $d, m \in \mathbf{Z}^+, r \in \mathbf{N}$ *be such that* $d \leq m$. *Let*

$$S \subseteq \{0, ..., r\}^m$$

such that S has G-, P- or GP-dimension no greater than d. Then

$$|S| \leq \sum_{i=0}^{d} \binom{m}{i} r^i.$$

Proof: Follows from Theorem 49 by substituting r for each r_k and collecting terms. □

Using similar techniques, we can establish the following.

Theorem 51: *For all* $d, m \in \mathbf{Z}^+, r_1, ..., r_m \in \mathbf{N}$ *such that* $d \leq m$,

$$\begin{aligned} \sum_{i=0}^{d} \sum_{S \in \Gamma_{m,i}} \prod_{k \in S} r_k &\leq N_{\max}(d, m, r_1, ..., r_m) \\ &\leq \sum_{i=0}^{d} \sum_{S \in \Gamma_{m,i}} \prod_{k \in S} \binom{r_k + 1}{2}. \end{aligned}$$

This gives the following improvement to Theorem 38.

Corollary 52: *Let* $d, m \in \mathbf{Z}^+, r \in \mathbf{N}$ *be such that* $d \leq m$. *Let*

$$S \subseteq \{0, ..., r\}^m$$

such that S has N-dimension no greater than d. Then

$$|S| \leq \sum_{i=0}^{d} \binom{m}{i} \binom{r + 1}{2}^i$$

Note that both Corollary 50 and Corollary 52 give Sauer's result (Theorem 32) in the case $r = 1$.

$$\psi_{GP,k}(i) = \begin{cases} 1 & \text{if } i = k \\ 0 & \text{if } i < k \\ \star & \text{if } i > k, \end{cases}$$

with a corresponding definition of the GP-dimension. The pseudo-dimension of S is denoted by P-dim(S), its Graph-dimension by G-dim(S), its GP-dimension by GP-dim(S) and its Natarajan dimension by N-dim(S).

Define

$$\begin{aligned} P_{\max}(d, m, r_1, ..., r_m) &= \max\{|S| : S \subseteq \prod_{i=1}^{m} \{0, ..., r_i\}, \text{P-dim}(S) \leq d\} \\ G_{\max}(d, m, r_1, ..., r_m) &= \max\{|S| : S \subseteq \prod_{i=1}^{m} \{0, ..., r_i\}, \text{G-dim}(S) \leq d\} \\ GP_{\max}(d, m, r_1, ..., r_m) &= \max\{|S| : S \subseteq \prod_{i=1}^{m} \{0, ..., r_i\}, \text{GP-dim}(S) \leq d\} \\ N_{\max}(d, m, r_1, ..., r_m) &= \max\{|S| : S \subseteq \prod_{i=1}^{m} \{0, ..., r_i\}, \text{N-dim}(S) \leq d\}. \end{aligned}$$

It is easily verified that if a set S N-shatters a set, it also GP-shatters it, and if S GP-shatters a set, it also G-shatters it and P-shatters it. This implies that

$$\begin{aligned} \text{N-dim}(S) &\leq \text{GP-dim}(S) \\ \text{GP-dim}(S) &\leq \text{P-dim}(S) \\ \text{GP-dim}(S) &\leq \text{G-dim}(S) \end{aligned}$$

which in turn implies that

$$\begin{aligned} P_{\max}(d, m, r_1, ..., r_m) &\leq GP_{\max}(d, m, r_1, ..., r_m) \\ G_{\max}(d, m, r_1, ..., r_m) &\leq GP_{\max}(d, m, r_1, ..., r_m) \\ GP_{\max}(d, m, r_1, ..., r_m) &\leq N_{\max}(d, m, r_1, ..., r_m) \end{aligned}$$

for all relevant $d, m \in \mathbf{Z}^+, r_1, ..., r_m \in \mathbf{N}$.

Our main result is stated below, and will be proved in the following section. In the following, for each $i, m \in \mathbf{Z}^+$, let $?_{m,i} \subseteq 2^{[m]}$ be defined by

$$?_{m,i} = \{S \subseteq [m] : |S| = i\}.$$

6. A Generalization of Sauer's Lemma

In this chapter, we improve the bounds of Theorem 38, and prove bounds of a similar flavor for other generalizations of the VC-dimension, including Natarajan's graph dimension [Natarajan, 1989] and Pollard's pseudo-dimension [Pollard, 1990]. In the latter two cases, we provide matching lower bounds. Next, we apply these bounds to the problem of bounding the uniform rate of convergence of empirical estimates to true means for families of (continuous-valued) random variables.

6.1 Statement of results

Let $[0] = \emptyset$ and for each $m \in \mathbf{N}$, let $[m]$ be the set $\{1, ..., m\}$.

Following [Bondy, 1972], let $(m, k) \rightarrow (n, l)$ denote the statement: If $S \subseteq \{0,1\}^m$, $|S| = k$, then there exists $\vec{q} \in \{0,1\}^m$ such that $\vec{q}$ has n 1's and

$$|\{\vec{s} \wedge \vec{q} : \vec{s} \in S\}| \geq l,$$

where $\wedge$ is the "bitwise AND" operation. Sauer's result can now be stated as

$$\left(m, 1 + \sum_{i=0}^{d-1} \binom{m}{i}\right) \rightarrow (d, 2^d).$$

Proofs of other statements of the form $(m, k) \rightarrow (n, l)$ and related results are given in [Anstee, 1985] [Anstee, 1991] [Anstee and Furedi, 1986] [Bondy, 1972] [Frankl *et al.*, 1987] [Frankl, 1983] [Tomasta, 1981].

Let $m \in \mathbf{Z}^+$. Let $r_i \in \mathbf{N}, 1 \leq i \leq m$. Let

$$S \subseteq X = \prod_{i=1}^{m} \{0, ..., r_i\}.$$

For $\vec{s} \in X$, denote by s_i the ith component of $\vec{s}$, and similarly for all cartesian products used in the chapter.

For the purpose of bounding the cardinality of sets of a given pseudo-dimension, and of a given Graph dimension, we define $GP = \{\psi_{GP,k} : k \in \mathbf{N}\}$, and

$a \in \{0, ..., r\}$, $l(a, a) = 0$, and if $a, b \in \{0, ..., r\}$ satisfy $a \neq b$, then $l(a, b) > 0$. Then we might ask that a learning strategy return a hypothesis $h \in \mathcal{F}$ such that the expected value of $l(h(x), y)$ is with high probability at most ϵ greater than the minimum of this expectation for all $f \in \mathcal{F}$. Note that the results of this paper can be described in this form using l_{discrete} where $l_{\text{discrete}}(a, b)$ is defined to be 0 when $a = b$ and 1 otherwise. For fixed r, the domain of any loss function l is finite. This implies that any loss function l satisfying the restrictions described above is always within a constant of l_{discrete}. This observation may be turned into a proof that $\mathcal{F}$ is learnable with respect to l in the aforementioned sense exactly when $\mathcal{F}$ is learnable with respect to l_{discrete}.

We may apply this observation to add the statement "$\mathcal{F}$ is uniformly convergent" to the list of equivalencies in Theorem 45. First, if $\mathcal{F}$ is uniformly convergent, it is learnable with respect to the loss function l_{abs} defined by $l_{\text{abs}}(a, b) = |a - b|$, by the results of Haussler [Haussler, 1991], and therefore it is learnable in the sense of this paper. Also, if $\mathcal{F}$ has finite P dimension, $\mathcal{F}$ is uniformly convergent, by the results of Pollard [Pollard, 1984].

Finally, the results of this section may be generalized to the "agnostic" generalization of this learning model (c.f., [Haussler, 1991]), where it is not assumed that the (x, y) pairs seen by the learner always satisfy $y = f(x)$, and the goal of the learner is to model the stochastic relationship between randomly drawn elements X and $\{0, ..., r\}$ nearly as well as possible, using functions in $\mathcal{F}$.

Say that a family Ψ of functions from $\{0,...,r\}$ to $\{0,1,*\}$ provides a characterization of learnability if and only if for any family $\mathcal{F}$ of $\{0,...,r\}$-valued functions the learnability of $\mathcal{F}$ is equivalent to the finiteness of either its Ψ-dimension or its uniform Ψ-dimension.

Theorem 47: *A family Ψ of functions from $\{0,...,r\}$ to $\{0,1,*\}$ provides a characterization of learnability if and only if Ψ is a distinguisher.*

Proof: Follows immediately from Theorem 45 and Lemma 46. □

Finally, we relate the learnability of $\{0,...,r\}$-valued functions to the learnability of $\{0,1\}$-valued functions. Intuitively, the problem of learning a class of $\{0,...,r\}$ valued functions reduces to $r+1$ problems of learning $\{0,1\}$-valued functions as follows. For $k=0,\ldots,r$ define $\mathcal{C}_k = \{c_{f,k} : f \in \mathcal{F}\}$ as the class of $\{0,1\}$-valued functions on X defined by

$$c_{f,k}(x) = \begin{cases} 1 & \text{if } f(x) = k \\ 0 & \text{otherwise.} \end{cases}$$

We can easily relate the learnability of $\mathcal{F}$ to the learnability of the $\mathcal{C}_k$'s as follows.

Theorem 48: *$\mathcal{F}$ is learnable if and only if $\mathcal{C}_k$ is learnable for each $k \subseteq \{0,...,r\}$.*

Proof: We claim that the uniform G-dimension of $\mathcal{F}$ is finite if and only if the VC-dimension of $\mathcal{C}_k$ is finite for all k. Suppose that $\mathcal{F}$ uniformly G-shatters a sequence $\vec{x}$ using ψ_k, then $\mathcal{C}_k$ clearly VC-shatters the same sequence.

The converse is also easily seen to hold. Since $\mathcal{F}$ can uniformly shatter a sequence if there exists at least a ψ_k such that the sequence is shattered using ψ_k, then a sequence is uniformly Ψ-shattered by $\mathcal{F}$ if and only if there exists at least a k such that the sequence is VC-shattered by $\mathcal{C}_k$. This proves the claim.

By the results of [Blumer *et al.*, 1989,Haussler, 1991], for each k, $\mathcal{C}_k$ is learnable exactly when is has finite VC-dimension. This completes the proof. □

Note that the results of this section can be trivially applied to obtain results in a more general model, in which certain errors are more serious than others. Suppose we defined a loss function l from $\{0,...,r\} \times \{0,...,r\}$ to the nonnegative reals such that for all

Since we require that the same M is sufficient for all distributions P, this is sometimes called *distribution free* uniform convergence.

Now we are ready for our main result which shows a variety of ways in which learnability can be characterized.

Theorem 45: *For any distinguisher Ψ, the following are equivalent:*

1. *The Ψ-dimension of $\mathcal{F}$ is finite.*
2. *The uniform Ψ-dimension of $\mathcal{F}$ is finite.*
3. *$L_{\mathcal{F}}$ is uniformly convergent.*
4. *The VC-dimension of $L_{\mathcal{F}}$ is finite.*
5. *$\mathcal{F}$ is learnable.*

Proof: Corollary 41 implies that (1. $\Leftrightarrow$ 2.). Theorem 43 implies that (5. $\Rightarrow$ 1.). Lemma 44 and Corollary 41 imply that (1. $\Leftrightarrow$ 4.). The implication (4. $\Rightarrow$ 3.) is an immediate consequence of the results in [Vapnik and Chervonenkis, 1971] and the implication (3. $\Rightarrow$ 5.) is a special case of [Haussler, 1991, Lemma 1, p. 20]. This completes the proof. □

The concept of distinguisher is a kind of metacharacterization, as it characterizes those Ψ which in turn characterize learnability, both through the finiteness of the Ψ-dimension, and through the finiteness of the uniform Ψ-dimension. To see this, all that remains is to show that for any family Ψ of functions from $\{0, ..., r\}$ to $\{0, 1, *\}$ which is not a distinguisher, neither the Ψ-dimension nor the uniform Ψ-dimension characterizes learnability.

Lemma 46: *If Ψ is a family of functions from $\{0, ..., r\}$ to $\{0, 1, *\}$ which is not a distinguisher, and if X is infinite, then there is a family $\mathcal{F}$ of functions from X to $\{0, ..., r\}$ which has Ψ-dimension 0 and has uniform Ψ-dimension 0, but which is not learnable.*

Proof: Suppose Ψ fails to distinguish $a_1, a_2 \in \{0, ..., r\}$. Then the set of all functions from X to $\{a_1, a_2\}$ trivially has Ψ-dimension and uniform Ψ-dimension 0. However, this class is trivially isomorphic to the set of all $\{0, 1\}$-valued functions defined on X, which was shown in [Blumer *et al.*, 1989] to not be PAC-learnable if X is infinite, so this class is trivially not learnable in this stronger setting. □

Proof: Suppose the sequence $x_1, ..., x_k$ of elements of X are G shattered by $\mathcal{F}$. Let $\psi_1, ..., \psi_k \in G$ be such that

$$\{(\psi_1(f(x_1)), ..., (\psi_k(f(x_k)))\} = \{0,1\}^k.$$

Let $a_1, ..., a_k \in \{0, ..., r\}$ be such that for all $j, 1 \leq j \leq k$, ψ_j is defined by

$$\psi_j(b) = \begin{cases} 1 & \text{if } b = a_j \\ 0 & \text{otherwise.} \end{cases}$$

Such a sequence $a_1, ..., a_k$ exists due to the definition of G-shattering. We claim that the sequence $(x_1, a_1), ..., (x_k, a_k)$ of elements of $X \times \{0, ..., r\}$ is (VC) shattered by $L_{\mathcal{F}}$. Choose $\vec{b} \in \{0,1\}^k$. Let $f \in \mathcal{F}$ be such that

$$\vec{b} = (\psi_1(f(x_1)), ..., \psi_k(f(x_k))).$$

Since, by definition, for all $j, 1 \leq j \leq k$, $l_f(x_j, a_j) = \psi_j(f(x_j))$, we have

$$\vec{b} = (l_f(x_1, a_1), ..., l_f(x_k, a_k)).$$

Since $\vec{b}$ was chosen arbitrarily, $L_{\mathcal{F}}$ shatters $(x_1, a_1), ..., (x_k, a_k)$. Thus, the VC-dimension of $L_{\mathcal{F}}$ is at least the graph dimension of $\mathcal{F}$.

Now, assume that a sequence $(x_1, a_1), ..., (x_k, a_k)$ of elements of $X \times \{0, ..., r\}$ is shattered by $L_{\mathcal{F}}$. We claim that $x_1, ..., x_k$ is G-shattered by $\mathcal{F}$. Define $\psi_1, ..., \psi_k \in G$, by

$$\psi_j(b) = \begin{cases} 1 & \text{if } b = a_j \\ 0 & \text{otherwise.} \end{cases}$$

Applying the fact that for all $j, 1 \leq j \leq k, l_f(x_j, a_j) = \psi_j(f(x_j))$, in a similar manner to the above verifies that $x_1, ..., x_k$ is G-shattered by $\mathcal{F}$, and therefore that the graph dimension of $\mathcal{F}$ is at least the VC-dimension of $L_{\mathcal{F}}$. This completes the proof. □

We say that $L_{\mathcal{F}}$ is *uniformly convergent* if for all $\epsilon > 0$, there is an $M \in \mathbf{N}$ such that for all $m \geq M$, for all probability measures P over $X \times \{0, ..., r\}$,

$$P^m \left\{ ((x_1, a_1), \ldots, (x_m, a_m)) : \exists f \in \mathcal{F}, \left| \frac{1}{m} \sum_{j=1}^{m} l_f(x_j, a_j) - P\{(x, a) : f(x) \neq a\} \right| \geq \epsilon \right\} \leq \epsilon.$$

4. The uniform Φ-dimension of $\mathcal{F}$ is infinite.

We make use of the following theorem of Natarajan, obtained trivially from a result of Ehrenfeucht, Haussler, Kearns and Valiant.

Theorem 42 ([Ehrenfeucht *et al.*, 1989,Natarajan, 1989]): *If the Natarajan dimension of $\mathcal{F}$ is infinite then $\mathcal{F}$ is not learnable.*

Combining Theorem 42 with Corollary 41, we obtain the following.

Theorem 43: *Let Ψ be a distinguisher. If $\mathcal{F}$ has infinite Ψ-dimension, then $\mathcal{F}$ is not learnable.*

Proof: By Corollary 41, if $\mathcal{F}$ has infinite Ψ-dimension, $\mathcal{F}$ has infinite Natarajan dimension. Applying Corollary 42 gives the desired result. □

Note that due to the correspondence between the indices of the vectors in $\mathcal{F}_{|\vec{x}}$ and elements of the domain X, the following definition of the Ψ-dimension of $\mathcal{F}$ is equivalent to that given at the beginning of this section. We say that a finite sequence $x_1, ..., x_k$ of elements of X is Ψ-shattered if there is a sequence $\psi_1, ..., \psi_k$ of elements of Ψ such that

$$\{(\psi_1(f(x_1)), ..., \psi_k(f(x_k))) : f \in \mathcal{F}\} \supseteq \{0,1\}^k$$

and let the Ψ-dimension of $\mathcal{F}$ be the length of the longest finite sequence shattered by $\mathcal{F}$, or infinity if arbitrarily long sequences of elements of X are shattered. We may also make the corresponding alteration to the definition of the uniform Ψ-dimension of $\mathcal{F}$. In the following lemma, we will find it convenient to use the altered definitions.

If f is a function from X to $\{0, ..., r\}$, define the function l_f from $X \times \{0, ..., r\}$ to $\{0,1\}$ by

$$l_f(x,a) = \begin{cases} 1 & \text{if } f(x) = a \\ 0 & \text{otherwise.} \end{cases}$$

Define $L_{\mathcal{F}} = \{l_f : f \in \mathcal{F}\}$.

Lemma 44: *The VC-dimension of $L_{\mathcal{F}}$ equals the graph dimension of $\mathcal{F}$.*

(2. $\Rightarrow$ 4.): This follows immediately from (1. $\Rightarrow$ 3.).

Finally, (3. $\Rightarrow$ 1.) and (4. $\Rightarrow$ 2.) follow immediately from Lemma 33. This completes the proof. $\Box$

5.2 Applications to learning

In this section, we describe applications of the results of the previous section to learning.

Choose a set X, a positive integer r and a family $\mathcal{F}$ of $\{0, ..., r\}$-valued functions defined on X. For a probability measure P over X we define the error of h with respect to f with respect to P, denoted by $\mathbf{er}_{f,P}(h)$ to be

$$P\{x : f(x) \neq h(x)\}.$$

A *learning strategy* for $\mathcal{F}$ is a mapping from finite sequences of elements of $X \times \{0, ..., r\}$ to $\mathcal{F}$. We say that $\mathcal{F}$ is *learnable* if there exists a learning strategy A and an integer-valued function $m(\varepsilon, \delta)$ such that for any $\varepsilon, \delta > 0$, for any probability measure P over X, and for any $f \in \mathcal{F}$, the probability that for $\vec{v} \in X^{m(\varepsilon,\delta)}$ drawn according to $P^{m(\varepsilon,\delta)}$ that

$$\mathbf{er}_{f,P}(A((v_1, f(v_1)), ..., (v_m, f(v_m)))) \leq \varepsilon$$

is at most δ. This definition of learnability is essentially that studied by Natarajan [Natarajan, 1989], and is based on Valiant's PAC model [Valiant, 1984]. We refer the interested reader to these papers for motivation.

Recall that at the end of Chaper 4, we extended the definition of Ψ-dimension from sets of vectors to sets of functions.

The results of the previous section immediately yield the following.

Corollary 41: *Choose distinguishers Ψ and Φ. Then the following are equivalent:*

1. *The Ψ-dimension of $\mathcal{F}$ is infinite.*
2. *The Φ-dimension of $\mathcal{F}$ is infinite.*
3. *The uniform Ψ-dimension of $\mathcal{F}$ is infinite.*

We are now ready for the main result of this section. It follows relatively straightforwardly from Lemma 33, Corollary 36 and Theorem 39.

Theorem 40: *Choose distinguishers Ψ and Φ. Then the following are equivalent:*

1. *The Ψ-dimension of S is infinite.*
2. *The Φ-dimension of S is infinite.*
3. *The uniform Ψ-dimension of S is infinite.*
4. *The uniform Φ-dimension of S is infinite.*

Proof: (1. $\Rightarrow$ 2.): Assume for contradiction that the Ψ-dimension S is infinite and the Φ-dimension is finite. Let d be the Φ-dimension of S. Choose k such that

$$\frac{k}{2\log(r+1)+\log k} > d.$$

Let m_1 and $\vec{\imath} = (i_1, ..., i_{m_1})$ be such that that Φ-dimension of $S_{|_{\vec{\imath}}}$ is d. Let m_2 and $\vec{\jmath} = (j_1, ..., j_{m_2})$ be such that that Ψ-dimension of $S_{|_{\vec{\jmath}}}$ is at least k. Let $\vec{z} = (i_1, ..., i_{m_1}, j_1, ..., j_{m_2})$. Let $S = S_{|_{\vec{z}}}$. Trivially, the Φ-dimension of S is d and the Ψ-dimension d' of S satisfies

$$\frac{d'}{2\log(r+1)+\log d'} > d. \tag{5.1}$$

Let d_N be the Natarajan dimension of S and let d_B be the B-dimension of S. Applying Corollary 36, we have that

$$\frac{d_B}{2\log(r+1)+\log d_B} > d_N,$$

but by Theorem 39, this is a contradiction.

(2. $\Rightarrow$ 1.): This follows from (1. $\Rightarrow$ 2.) by symmetry.

(1. $\Rightarrow$ 3.): Assume for contradiction that the Ψ-dimension S is infinite and the uniform Ψ-dimension is finite. Let d be the uniform Ψ-dimension of S. Let m_1 and $\vec{\imath} = (i_1, ..., i_{m_1})$ be such that that uniform Ψ-dimension of $S_{|_{\vec{\imath}}}$ is d. Let m_2 and $\vec{\jmath} = (j_1, ..., j_{m_2})$ be such that that Ψ-dimension of $S_{|_{\vec{\jmath}}}$ is greater than $|\Psi|d$. Let $\vec{z} = (i_1, ..., i_{m_1}, j_1, ..., j_{m_2})$. Let $S = S_{|_{\vec{z}}}$. Again, trivially, the uniform Ψ-dimension of S is d and the Ψ-dimension of S is greater that $|\Psi|d$, which is a contradiction.

Theorem 37: *Choose a set Ψ of functions from $\{0,...,r\}$ to $\{0,1,*\}$. If S has uniform Ψ-dimension at most d, then it has Ψ-dimension at most $d|\Psi|$.*

Proof: Suppose that the Ψ-dimension d' of S is greater than $d|\Psi|$. Let $\vec{\imath} = (i_1, ..., i_{d'})$ be a sequence shattered by S, and let $\vec{\psi} = (\psi_1, ..., \psi_{d'})$, be such that

$$\{0,1\}^{d'} \subseteq S_{|\vec{\imath}}.$$

By the pigeonhole principle, since $d' > d|\Psi|$, there exists a subsequence $(i_{j_1}, \ldots, i_{j_{d+1}})$ of $\vec{\imath}$ such that for all $1 \leq k, l \leq d+1$, $\psi_{j_k} = \psi_{j_l}$. Therefore, S uniformly Ψ-shatters $(i_{j_1}, \ldots, i_{j_{d+1}})$, contradicting the assumption that the uniform Ψ-dimension of S is at most d. □

We will make use of a theorem of Natarajan.[3]

Theorem 38 ([Natarajan, 1989]): *If the Natarajan dimension of S is at most d, then*

$$|S| \leq m^d (r+1)^{2d}.$$

We may apply this theorem to obtain lower bounds on the Natarajan dimension in terms of the B-dimension.

Theorem 39: *Let $S \subseteq \{0,...,r\}^m$. Let d_N be the Natarajan dimension of S and d_B be the B-dimension of S. Then,*

$$d_N \geq \frac{d_B}{2\log(r+1) + \log d_B}.$$

Proof: Let $\vec{\imath} = (i_1, ..., i_{d_B})$ be a sequence of indices B-shattered by S. Let $T = S_{|\vec{\imath}}$. Since there exists $\psi \in B$ such that

$$\{0,1\}^{d_B} \subseteq \psi(T),$$

we have that $|T| \geq 2^{d_B}$. From Theorem 38, we may conclude that $|T| \leq d_B^{d_N}(r+1)^{2d_N}$. Thus,

$$d_B^{d_N}(r+1)^{2d_N} \geq 2^{d_B}.$$

Taking logs and solving for d_N yields the desired result.□

[3]The bounds of this theorem are improved in Chapter 6, but Natarajan's result is sufficient for the purposes of this chapter.

Thus S Φ-shatters $\vec{\imath}$. The uniform case follows analogously. $\Box$

Let Ψ be a family of functions from $\{0, ..., r\}$ to $\{0, 1, *\}$. We say that a pair a, b of distinct elements in $\{0, ..., r\}$ is Ψ-*distinguishable* if there exists $\psi \in \Psi$ such that $\psi(a) = 0$ and $\psi(b) = 1$ or vice versa. We say Ψ is a *distinguisher* if each pair $a, b \in \{0, ..., r\}$ is Ψ-distinguishable. It is easy to see that in the case $r = 1$, for any distinguisher Ψ, the definitions of the Ψ-dimension and the uniform Ψ-dimension are equivalent to the definition of the VC-dimension.

Next, we describe a certain sense in which B is the maximum of the set of Ψ's which are distinguishers and N is the minimum.

Theorem 35: *Choose a distinguisher Ψ. Choose $S \in \{0, ..., r\}^m$, and choose $\vec{\imath} \in \{1, ..., m\}^k$.*

- *If S N-shatters $\vec{\imath}$, then S Ψ-shatters $\vec{\imath}$.*
- *If S Ψ-shatters $\vec{\imath}$, then S B-shatters $\vec{\imath}$.*
- *If S uniformly N-shatters $\vec{\imath}$, then S uniformly Ψ-shatters $\vec{\imath}$.*
- *If S uniformly Ψ-shatters $\vec{\imath}$, then S uniformly B-shatters $\vec{\imath}$.*

Proof: Follows immediately from Lemma 34 and the definition of a distinguisher. $\Box$

This theorem trivially yields the following Corollary about the Ψ-dimension and the uniform Ψ-dimension for various Ψ.

Corollary 36: *Choose a distinguisher Ψ and $S \in \{0, ..., r\}^m$.*

- *The Natarajan dimension of S is at most the Ψ-dimension of S.*
- *The Ψ-dimension of S is at most the B-dimension of S.*
- *The uniform Natarajan dimension of S is at most the uniform Ψ-dimension of S.*
- *The uniform Ψ-dimension of S is at most the uniform B-dimension of S.*

Next, a simple pigeonhole argument establishes the following bound on the uniform Ψ-dimension of S in terms of its Ψ-dimension, for any Ψ.

5.1 Generalizations of the VC-dimension

Since in this chapter, we will be concerned with learning $\{0,...,r\}$-valued functions for fixed r, we will restrict our attention in this section to the Ψ-dimension of subsets of $\{0,...,r\}^m$, retreating a little from the generality of the introduction. Note that when we fix r, instead of considering classes Ψ of functions from $\mathbf{N}$ to $\{0,1,*\}$, we may restrict our attention to classes of functions from $\{0,...,r\}$ to $\{0,1,*\}$, since the behavior of functions in Ψ outside $\{0,...,r\}$ does not affect the Ψ-dimension of a given subset of $\{0,...,r\}^m$. For the specific notions of dimension described in the introduction, we obtain identical definitions by simply restricting the functions in Ψ to $\{0,...,r\}$.

We begin by describing a sufficient condition for Ψ-shattering to imply Φ-shattering.[2]

Lemma 34: *Let Ψ, Φ be classes of functions from $\{0,...,r\}$ to $\{0,1,*\}$ such that for all $\psi \in \Psi$ there exists $\phi \in \Phi$, such that $\psi^{-1}(0) \subseteq \phi^{-1}(b)$ and $\psi^{-1}(1) \subseteq \phi^{-1}(1-b)$ holds for b either 0 or 1. Then for all $S \subseteq \{0,...,r\}^m$, $\vec{\imath} \in \{1,...,m\}^k$, if S Ψ-shatters $\vec{\imath}$, then S Φ-shatters $\vec{\imath}$, and if S uniformly Ψ-shatters $\vec{\imath}$, then S uniformly Φ-shatters $\vec{\imath}$.*

Proof: Assume that for all $\psi \in \Psi$, there is a $\phi \in \Phi$ such that $\psi^{-1}(0) \subseteq \phi^{-1}(0)$ and $\psi^{-1}(1) \subseteq \phi^{-1}(1)$ (the case in which for all $\psi \in \Psi$, there is a $\phi \in \Phi$ such that $\psi^{-1}(0) \subseteq \phi^{-1}(1)$ and $\psi^{-1}(1) \subseteq \phi^{-1}(0)$ can be handled analogously). Choose $S \subseteq \{0,...,r\}^m$, $\vec{\imath} \in \{1,...,m\}^k$ such that S Ψ-shatters $\vec{\imath}$. Choose $\vec{\psi} \in \Psi^k$ such that

$$\{0,1\}^k \subseteq \vec{\psi}(S_{|\vec{\imath}}).$$

For each $j, 1 \leq j \leq k$, let ϕ_j be such that $\psi_j^{-1}(0) \subseteq \phi_j^{-1}(0)$ and $\psi_j^{-1}(1) \subseteq \phi_j^{-1}(1)$. Let $\vec{\phi} = (\phi_1,...,\phi_k)$.

We claim that $\{0,1\}^k \subseteq \vec{\phi}(S_{|\vec{\imath}})$. Choose $\vec{b} = (b_1,...,b_k) \in \{0,1\}^k$. Let $\vec{q} \in S_{|\vec{\imath}}$ be such that $\vec{\psi}(\vec{q}) = \vec{b}$. Choose $j \in \{1,...,k\}$. Since $\psi_j^{-1}(0) \subseteq \phi_j^{-1}(0)$, and $\psi_j^{-1}(1) \subseteq \phi_j^{-1}(1)$, and $b_j \in \{0,1\}$, $\phi_j(q_j) = \psi_j(q_j)$. Since j was chosen arbitrarily, $\vec{\phi}(\vec{q}) = \vec{\psi}(\vec{q}) = \vec{b}$. Therefore, since $\vec{b}$ was chosen arbitrarily,

$$\{0,1\}^k \subseteq \vec{\phi}(S_{|\vec{\imath}}).$$

[2]The definition of Ψ-shattering is given at the beginning of this part.

5. Characterizations of Learnability for Classes of Many-valued Functions

A central task in the theory of computational learning is to provide a simple mathematical characterization of the classes of concepts that are learnable under some formal model of learning. An example along these lines is the characterization of Valiant's PAC-learnability of binary functions in terms of the Vapnik-Chervonenkis dimension[1] proved by Blumer, Ehrenfeucht, Haussler and Warmuth [Blumer *et al.*, 1989].

A natural way to extend that model is to consider the learning of many-valued (instead of binary) functions. A characterization of PAC-learnability for classes of many-valued functions has been obtained by Natarajan in terms of a particular generalization of the VC-dimension which we will call the Natarajan dimension [Natarajan, 1989].

In this chapter we introduce a general scheme for extending the VC-dimension to classes of $\{0, ..., r\}$-valued functions. This scheme gives rise to a wide family of notions of the dimension of a class of functions. Our family of generalizations of the VC-dimension includes as special cases the Natarajan dimension [Natarajan, 1989], the graph dimension [Dudley, 1987,Natarajan, 1989], Pollard's pseudo-dimension [Pollard, 1984,Pollard, 1990, Haussler, 1991], and a generalization proposed by Vapnik (see, e.g. [Vapnik, 1989]).

By establishing the existence of a minimum and a maximum in the family of generalizations and proving that the finiteness of both these dimensions is equivalent, we obtain a variety of clear combinatorial characterizations of PAC-learnability for classes of multi-valued functions.

This research provides more flexible tools for determining whether a class of $\{0, ..., r\}$-valued functions is learnable, and enhances the understanding of why the finiteness of the previously studied generalizations of the VC-dimension characterize learnability.

[1] Defined by Vapnik and Chervonenkis [Vapnik and Chervonenkis, 1971].

The Natarajan dimension [Natarajan, 1989] is the N-dimension where $N = \{\psi_{N,k,l} : k, l \in \mathbf{N}, k \neq l\}$ and $\psi_{N,k,l}$ is defined by

$$\psi_{N,k,l}(a) = \begin{cases} 1 & \text{if } a = k \\ 0 & \text{if } a = l \\ * & \text{otherwise.} \end{cases}$$

Finally, let B be the set of all functions from $\mathbf{N}$ to $\{0, 1\}$ and define the B-dimension accordingly.

Note that the graph dimension, the Natarajan dimension and the B-dimension do not make use of the natural ordering on $\{0, ..., r\}$ and could just as easily be defined for abstract finite sets.

For any (possibly infinite) set X, $r \geq 1$, any class Ψ of functions from $\{0, ..., r\}$ to $\{0, 1, *\}$, and any class $\mathcal{F}$ of functions from X to $\{0, ..., r\}$, define the Ψ-dimension of $\mathcal{F}$ to be maximum, over all finite sequences $x_1, ..., x_m$ of elements of X, of the Ψ-dimension of

$$\{(f(x_1), ..., f(x_m)) : f \in \mathcal{F}\},$$

if such a maximum exists, and infinity otherwise.

We say $\vec{\imath}$ is Ψ-shattered by S if there exists $\vec{\psi} \in \Psi^k$ such that

$$\{0,1\}^k \subseteq \vec{\psi}(S_{|\vec{\imath}}).$$

We say that $\vec{\psi}$ *witnesses* this shattering. Furthermore, for any $\vec{b} \in \{0,1\}^k$, an $\vec{s} \in S$ for which $\vec{\psi}(s_{i_1}, ..., s_{i_k}) = \vec{b}$ is said to *satisfy* $\vec{b}$ (with respect to $\vec{\imath}$; we will often let $\vec{\imath}$ be understood from context).

In the case in which there exists such a $\vec{\psi}$ which in addition has $\psi_1 = \psi_2 = \cdots = \psi_k$, we say that $\vec{\imath}$ is *uniformly* Ψ-shattered by S.

Let the Ψ-dimension of S be the maximum d for which there exists a sequence $\vec{\imath} \in \{1, ..., m\}^d$ of indices shattered by S, and let the uniform Ψ-dimension of S be the corresponding definition for uniform shattering.[1]

We will make use of the following observation.

Lemma 33: *For any set Ψ of functions from $\{0, ..., r\}$ to $\{0, 1, *\}$, and any $S \subseteq \{0, ..., r\}^m$, the uniform Ψ-dimension of S is at most the Ψ-dimension of S.*

Several previously defined notions of dimension correspond to particular choices of the set Ψ as shown in the following list.

Pollard's pseudo-dimension [Pollard, 1990] is the P-dimension, where $P = \{\psi_{P,k} : k \in \mathbf{N}\}$, and $\psi_{P,k}$ is defined by

$$\psi_{P,k}(a) = \begin{cases} 1 & \text{if } a \geq k \\ 0 & \text{otherwise.} \end{cases}$$

Vapnik [Vapnik, 1989] describes a generalization of the VC-dimension which is equivalent to the uniform P-dimension.

The graph dimension [Dudley, 1987,Natarajan, 1989] is the G-dimension, where $G = \{\psi_{G,k} : k \in \mathbf{N}\}$, and $\psi_{G,k}$ is defined by

$$\psi_{G,k}(a) = \begin{cases} 1 & \text{if } a = k \\ 0 & \text{otherwise.} \end{cases}$$

[1] Notice that these definitions are the same as we would obtain if we insisted that the shattered indices satisfy $1 \leq i_1 < i_2 < \cdots < i_k \leq m$, which is perhaps the easiest way to think of this definition.

The VC-dimension of S is the length of the longest finite sequence of indices VC-shattered by S.

The following Theorem, often called Sauer's Lemma, will play a central role in this part. Vapnik and Chervonenkis independently proved a similar lemma.

Theorem 32 ([Sauer, 1972,Vapnik and Chervonenkis, 1971]): *Let $S \subseteq \{0,1\}^m$. If the VC-dimension of S is d, then*

$$|S| \leq \sum_{i=0}^{d} \binom{m}{i} \leq (em/d)^d$$

and the first bound is tight; i.e., for all $d, m \in \mathbf{Z}^+, d \leq m$, there exists $S \subseteq \{0,1\}^m$ of VC-dimension d that meets that upper bound.

Now, let us return to the more general case in which the r_i's may be greater than 1. A natural way to extend the above definition of shattering is to say that S shatters $\vec{\imath}$ if and only if

$$S_{|\vec{\imath}} = \prod_{j=1}^{k} \{0, ..., r_{i_j}\}$$

and define a notion of dimension as we did with the VC-shattering. Generalizations of Sauer's Lemma using this extension of the definition of shattering were described in [Alon, 1983] [Karpovsky and Milman, 1978] [Steele, 1978] [Anstee, 1988]. Unfortunately, using this latter definition, the cardinality of the largest subset of $\prod_{l=1}^{m} \{0, ..., r_l\}$ of a given dimension grows like 2^m, whereas slower (e.g., polynomial) growth is desirable for learning results, as we will see in Chapter 6.

To define generalizations that yield bounds polynomial in m, we look for a "translation" of multi-valued vectors into binary vectors. This is done by considering mappings ψ from $\mathbf{N}$ to $\{0, 1, *\}$ ($*$ will be thought of as a null element) which we will apply to the components of elements of S.

Let Ψ be a family of functions from $\mathbf{N}$ to $\{0, 1, *\}$. For $\vec{u} \in \{0, ..., r\}^m$, $\vec{\psi} = (\psi_1, ..., \psi_m) \in \Psi^m$, denote $(\psi_1(u_1), ..., \psi_m(u_m))$ by $\vec{\psi}(\vec{u})$. For a set $U \subseteq \{0, ..., r\}^m$, define $\vec{\psi}(U) = \{\vec{\psi}(\vec{u}) : \vec{u} \in U\}$.

functions in an extension of Valiant's model proposed by Haussler [Haussler, 1991].

In Chapter 7, we examine a variant of this model in which it is assumed that learning is an "on-line," never-ending process. Further, we assume that the function to be learned is ever changing, however slowly. Examples of natural functions which change slowly over time are "stylish clothing" ($f(x) = 1$ if x represents a "stylish" article of clothing), "polite conversation," and "obscene books." We prove upper and lower bounds on rate of change that a learner can tolerate in terms of the accuracy required of the learner's hypotheses, and we describe efficient algorithms for the learner in this setting for two commonly studied examples of classes $\mathcal{F}$ assumed to contain the hidden functions.

4.1 Some definitions

Once again, basic mathematical definitions and notation are given in Appendix A.

After Vapnik [Vapnik, 1989], we will adopt a naive attitude toward measurability, assuming that every set is measurable, and simply speak of probability distributions on sets.

The Vapnik-Chervonenkis (VC) dimension [Vapnik and Chervonenkis, 1971], and generalizations thereof, have been invaluable in analyzing the ability of computers to learn in a random environment (c.f., [Blumer *et al.*, 1989] [Haussler *et al.*, 1990]). We define the VC-dimension and some of its generalizations formally here, as they will be useful throughout this part.

Choose $m, r_1, ..., r_m \in \mathbf{N}$. Let $S \subseteq \prod_{l=1}^{m} \{0, ..., r_l\}$.

For a sequence $\vec{\imath} = (i_1, ..., i_k)$ of indices, where $1 \leq i_j \leq m$ for each $1 \leq j \leq k$, define $S_{|_{\vec{\imath}}} \subseteq \prod_{j=1}^{k} \{0, ..., r_{i_j}\}$ by

$$S_{|_{\vec{\imath}}} = \{(s_{i_1}, ..., s_{i_k}) : \vec{s} \in S\}.$$

Suppose for a moment that $r_1 = \cdots = r_m = 1$. In such a case, we say that S VC-shatters a finite sequence $\vec{\imath} = (i_1, ..., i_k)$ of indices if and only if

$$S_{|_{\vec{\imath}}} = \{0, 1\}^k.$$

4. Introduction

In this chapter, we will consider two models of learning in which probabilistic assumptions are made about the environment of the learner. Both are variants of Valiant's PAC ("Probably Approximately Correct") model [Valiant, 1984].

In Valiant's model, it is assumed that a $\{0,1\}$-valued function f is hidden from the learner, and that the learner know of a class $\mathcal{F}$ containing the hidden function f. The learner receives several examples $(x_1, f(x_1)), ..., (x_m, f(x_m))$ of the behavior of the hidden function f, and uses these examples to construct a hypothesis h, which it hopes is a good approximation to f. It is further assumed that an arbitrary, unknown probability distribution D on the domain of f was used to independently, randomly generate the x_i's. The inaccuracy of the learner's hypothesis h is then measured by the probability, according to D, that f and h yield different values when applied to yet another element of their domain generated according to D. This model demands that with high probability, with respect to the random x_i's, the learner's hypothesis obtained from those x_i's is very accurate. It further demands that the learner compute its hypothesis efficiently.

In Chapter 5, we consider a variant of Valiant's model in which the function to be learned may be many-valued. We unify several previous results which described simple tests to determine whether a class $\mathcal{F}$ of such many-valued functions is "learnable" or not with respect to a cousin of Valiant's PAC model, in which the computational effort expended by the learner is ignored. We can see these tests as special cases of a "testing scheme," which includes many more tests. Thus, the results of Chapter 5 provide additional tools for understanding the basics of learning many-valued functions, and we hope they aid understanding of the previously developed tools.

In Chapter 6, we generalize a well-known combinatorial theorem, often called "Sauer's Lemma." Sauer's Lemma has often been applied to problems of learning $\{0,1\}$-valued functions [Blumer *et al.*, 1989,Haussler *et al.*, 1990,Haussler *et al.*, 1991]. We show how our generalization may be applied to obtain results concerning the learning of real-valued

Part II

Learning in a Random Environment

Theorem 31: *Let* $\sigma = \langle x_t \rangle_{1 \leq t \leq m}$ *be any sequence of* m *elements of* $[0,1]$*, and let* $f \in \mathcal{F}_2$*. Then*

$$L_1(LININT, f, \sigma) \leq e(2 + \frac{\log m}{2}).$$

We have so far been unable to obtain matching lower bounds.

Both proofs make use of the following inequality, which follows immediately by the standard convexity argument.

Lemma 29: *For any* $n \in \mathbf{N}, p > 1, \vec{x} \in \mathbf{R}^n$,

$$||\vec{x}||_1 \leq n^{1-1/p}||\vec{x}||_p.$$

We begin with $\mathcal{F}_\infty$.

Theorem 30: *Choose* $m \geq 3$. *Let* $\sigma = \langle x_t \rangle_{1 \leq t \leq m}$ *be any sequence* m *examples elements of* $[0,1]$, *and choose* $f \in \mathcal{F}_\infty$. *Let* A *be the "nearest neighbor" algorithm. Then,*

$$L_1(A, f, \sigma) \leq e(1 + \frac{\log m}{2}).$$

Proof: Let $\lambda_1, ..., \lambda_m$ be the sequence of A's predictions. Let $\vec{r} \in \mathbf{R}^m$ be defined by

$$\vec{r} = (|\lambda_1 - f(x_1)|, ..., |\lambda_m - f(x_m)|).$$

Choose $p > 1$. By Theorem 27, we have

$$||\vec{r}||_p \leq \left[1 + \frac{1}{2^p - 2}\right]^{1/p}.$$

Applying Lemma 29, we have

$$||\vec{r}||_1 \leq m^{1-1/p}\left[1 + \frac{1}{2^p - 2}\right]^{1/p}.$$

Suppose $p = (\ln m)/(\ln m - 1)$. Then

$$\begin{aligned}
||\vec{r}||_1 &\leq m^{1-\frac{\ln m - 1}{\ln m}}\left[1 + \frac{1}{2^{(\ln m)/(\ln m - 1)} - 2}\right]^{\frac{\ln m - 1}{\ln m}} \\
&= e\left[1 + \frac{1}{2}\left(\frac{1}{\exp(\frac{\ln 2}{\ln m - 1}) - 1}\right)\right]^{1 - 1/\ln m} \\
&\leq e(1 + \frac{1}{2}\left(\frac{1}{\exp(\frac{\ln 2}{\ln m}) - 1}\right)) \\
&\leq e(1 + \frac{\ln m}{2\ln 2}) \qquad \text{(Lemma 26)} \\
&= e(1 + \frac{\log m}{2})
\end{aligned}$$

This completes the proof. □

With minor modifications, the above argument, together with Theorem 28, yields the following.

by (3.8). Also,

$$\begin{aligned}
\sum_{t>1:e_t>d_t} e_t^p &\leq \sum_{t>1:e_t>d_t} e_t \qquad \text{(Since } e_t \leq 1\text{)} \\
&< \sum_{t>1:e_t>d_t} e_t(e_t/d_t) \\
&= \sum_{t>1:e_t>d_t} e_t^2/d_t \\
&\leq 1,
\end{aligned}$$

by (3.7). Combining with (3.9) yields the first inequality. The second follows immediately from Lemma 26. □

3.5 Bounded-length trial sequences

In Section 3.2, we showed that $\mathrm{LC}_1(\mathcal{F}_\infty) = \mathrm{LC}_1(\mathcal{F}_2) = \infty$. In other words, we showed that finite bounds on the sum of absolute differences between predictions and true values could not be obtained for any algorithm using only the assumption that the hidden function was in $\mathcal{F}_\infty$, and therefore, for any algorithm using only the weaker assumption that the hidden function was in $\mathcal{F}_2$. Our adversaries used many trials, forcing small errors on each trial. The fact that $\mathrm{LC}_2 < \infty$ for both these classes suggests that this behavior was necessary, since, as the error on a trial approaches 1, squaring the error has no effect.

If, in fact, any adversary which forces infinite cumulative error for algorithms learning $\mathcal{F}_\infty$ must force small errors on each trial, this is good news for the learner, since, even if one's total error is unbounded, if it is accumulating slowly, nontrivial learning is taking place.

In this section, we show that, indeed, the "nearest neighbor" algorithm studied in the previous section accumulates error slowly while learning $\mathcal{F}_\infty$. We show that on any sequence of m trials consistent with a function in $\mathcal{F}_\infty$, the sum of unsquared errors made by the nearest neighbor algorithm is $O(\log m)$. We also show that the "linear interpolation" algorithm studied in Section 3.3 achieves the same bound on its cumulative (unsquared) error on any sequence of m trials consistent with a function in $\mathcal{F}_2$.

Proof: Consider the algorithm A which simply predicts with the function value at the nearest previously seen point (and arbitrarily on the first trial). Choose a sequence $x_1, ..., x_m$ of elements of $[0,1]$ and $f \in \mathcal{F}_\infty$. Let $\lambda_2, ..., \lambda_m$ be the predictions of this "nearest neighbor" algorithm on trials 2 through m. We have

$$\begin{aligned}\sum_{t=2}^{m} |\lambda_t - f(x_t)|^p &\leq \sum_{t=2}^{m} (\min_{i<t} |x_i - x_t|)^p \quad \text{(Lemma 25)} \\ &\leq 1 + \frac{1}{2^p - 2} \quad \text{(Lemma 24)}\end{aligned}$$

completing the proof of the first inequality of the theorem. The second follows immediately from Lemma 26. □

Next, we prove a very similar bound on $\mathrm{LC}_p(\mathcal{F}_2)$.

Theorem 28: *If $p > 1$,*

$$LC_p(\mathcal{F}_2) \leq 2 + \frac{1}{2^p - 2} \leq 2 + \frac{1}{(2\ln 2)(p-1)}.$$

Proof: Choose $p > 1$. Choose a sequence $x_1, ..., x_m$ of elements of $[0,1]$ and $f \in \mathcal{F}_\infty$. Let $\lambda_2, ..., \lambda_m$ be the predictions of LININT on trials 2 through m, and for each $t > 1$, let $d_t = \min_{i<t} |x_i - x_t|$, and let $e_t = |\lambda_t - f(x_t)|$. Applying Lemma 20, we have that the action of LININT's hypothesis increases by at least e_t^2/d_t on each trial. By Lemma 19, the action of LININT's hypothesis is always at most 1. Thus,

$$\sum_{t=2}^{m} e_t^2/d_t \leq 1. \tag{3.7}$$

Since, by Lemma 24, we have

$$\sum_{t=2}^{m} d_t^p \leq 1 + \frac{1}{2^p - 2}, \tag{3.8}$$

our analysis proceeds by breaking up the trials, and applying (3.7) to those trials where d_t is relatively small, and (3.8) to the trials where d_t is relatively large.

More specifically, we have

$$\begin{aligned}\sum_{t>1:e_t \leq d_t} e_t^p &\leq \sum_{t>1:e_t \leq d_t} d_t^p \\ &\leq 1 + \frac{1}{2^p - 2}, \end{aligned} \tag{3.9}$$

By differentiating, we may easily see that this expression, as a function of a, is decreasing when $a, d_t > 0$. Thus, it is maximized, subject to $a \geq 2d_t$, when $a = 2d_t$. Since $(2 - 2^p)d_t^p < 0$, this yields

$$H_t - H_{t-1} \leq (2 - 2^p)d_t^p.$$

Since, trivially, $0 \leq H_t \leq 1$ for all t, and H_t never increases (on any trial), we have (3.6). Combining (3.5) and (3.6) yields the desired bound. □

We begin with $\mathcal{F}_\infty$. We will make use of the following simple lemma, whose proof is omitted. It establishes the fact that functions in $\mathcal{F}_\infty$ satisfy a Lipschitz condition.

Lemma 25: *If $f \in \mathcal{F}_\infty$, then for all $x, y \in [0, 1]$, we have*

$$|f(x) - f(y)| \leq |x - y|.$$

We will also make use of the following approximation.

Lemma 26: *For all real x,*

$$\frac{1}{e^{1/x} - 1} \leq x.$$

Proof: We have

$$\begin{aligned} e^{1/x} &\geq 1 + 1/x \\ e^{1/x} - 1 &\geq 1/x \\ \frac{1}{e^{1/x} - 1} &\leq x. \end{aligned}$$

This completes the proof. □

A bound on $\mathrm{LC}_p(\mathcal{F}_\infty)$ follows immediately.

Theorem 27: *If $p > 1$,*

$$LC_p(\mathcal{F}_\infty) \leq 1 + \frac{1}{2^p - 2} \leq 1 + \frac{1}{(2 \ln 2)(p - 1)}.$$

Lemma 24: *Choose a sequence $x_1, x_2, ...$ of elements of $[0,1]$. For each $t > 1$, let*

$$d_t = \min_{i<t} |x_t - x_i|.$$

If $p > 1$,

$$\sum_{t=1}^{\infty} d_t^p \leq 1 + 1/(2^p - 2).$$

Proof: Choose a sequence $x_1, x_2, ...$ of elements of $[0,1]$. Assume without loss of generality that the x_i's are distinct. For each $t \in \mathbf{N}$, let

$$S_t = \{x_i : i \leq t\} = \{u_{i,t} : i \leq t\},$$

where $u_{1,t} < u_{2,t} < \cdots u_{t,t}$ (the $u_{i,t}$'s are $\{x_1, ..., x_t\}$ in sorted order). For each t, let $s_t = u_{t,t} - u_{1,t}$, and

$$H_t = u_{1,t}^p + (1 - u_{t,t})^p + \sum_{i=1}^{t-1} (u_{i+1,t} - u_{i,t})^p.$$

First, we claim that

$$\sum_{t>1 : x_t \notin [u_{1,t-1}, u_{t-1,t-1}]} d_t^p \leq 1. \tag{3.5}$$

Choose a trial t for which $x_t < u_{1,t-1}$. In such a case, we have

$$s_t - s_{t-1} = d_t \geq d_t^p$$

since $d_t \leq 1$ and $p > 1$. Similarly, if $x_t > u_{t-1,t-1}$, then $s_t - s_{t-1} \geq d_t^p$. Since, trivially, s_t never decreases, and $0 \leq s_t \leq 1$, we have (3.5).

Next, we claim that

$$\sum_{t : x_t \in [u_{1,t-1}, u_{t-1,t-1}]} d_t^p \leq 1/(2^p - 2). \tag{3.6}$$

Choose a trial t for which $x_t \in [u_{1,t-1}, u_{t-1,t-1}]$. Let i be such that $x_t \in (u_{i,t-1}, u_{i+1,t-1})$. Let $a = u_{i+1,t-1} - u_{i,t-1}$. Assume, as a first case, that x_t is closest to $u_{i,t-1}$ (the other case may be handled similarly). Then $d_t = x_t - u_{i,t-1} \leq a/2$. We have

$$H_t - H_{t-1} = d_t^p + (a - d_t)^p - a^p.$$

We may apply this result to obtain an alternative proof of a result of Faber and Mycielski [Faber and Mycielski, 1991], who analyzed another, more complicated, algorithm for their upper bounds.

Theorem 22 ([Faber and Mycielski, 1991]):

$$LC_2(\mathcal{F}_2) = 1.$$

Proof: The previous theorem implies that $\mathrm{LC}_2(\mathcal{F}_2) \leq 1$. To see that $\mathrm{LC}_2(\mathcal{F}_2) \geq 1$, consider an adversary which gives a first example of $(0, 0)$, and a second example of $(1, \pm 1)$, depending on whether an algorithm's prediction is positive or negative. This completes the proof. □

As discussed in the introduction, the fact that $\mathcal{F}_\infty \subseteq \mathcal{F}_2$, together with the same adversary argument as above, trivially yields the following.

Corollary 23: $LC_2(\mathcal{F}_\infty) = 1$.

This corollary tells us that, with respect to worst-case cumulative squared error, the assumption that the derivative of a hidden function is never more than 1 doesn't give the learner any more power than the assumption that the average value of the square of the derivative is at most one.[3]

3.4 More general loss functions

Recall that in Section 3.3, we proved that $\mathrm{LC}_2(\mathcal{F}_\infty) = \mathrm{LC}_2(\mathcal{F}_2) = 1$, and in Section 3.2, we proved that $\mathrm{LC}_1(\mathcal{F}_\infty) = \mathrm{LC}_1(\mathcal{F}_2) = \infty$. This brings up a natural question: For which p are $\mathrm{LC}_p(\mathcal{F}_\infty)$ and $\mathrm{LC}_p(\mathcal{F}_2)$ finite? This question is resolved in this section: we show that $\mathrm{LC}_p(\mathcal{F}_\infty)$ and $\mathrm{LC}_p(\mathcal{F}_2)$ are finite whenever $p > 1$.

The following lemma will be useful in both analyses.

[3]It would appear that the assumption that $f \in \mathcal{F}_\infty$ amounts to the slightly weaker assumption that the measure of $\{x : f'(x) > 1\}$ is zero. However, it is easy to see that the lower bound also applies to the smaller class of twice differentiable functions for which $f' \leq 1$ (indeed, to the extremely simple class consisting only of $f(x) = x$ and $g(x) = -x$). Thus, the square loss learning complexity of this class is the same as that of $\mathcal{F}_2$.

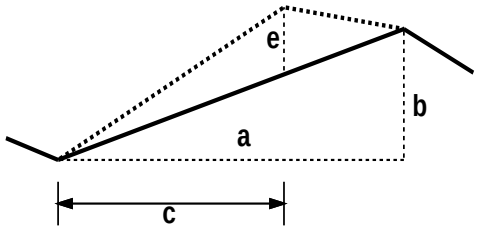

Figure 3.1: Change in J

Now we are ready for the learning result. Consider the learning algorithm LININT defined by

$$\text{LININT}(\emptyset, x_1) = 0$$

and

$$\begin{aligned} \lambda_t &= \text{LININT}(((x_1, y_1), ..., (x_{t-1}, y_{t-1})), x_t) \\ &= f_{\{(x_1,y_1),...,(x_{t-1},y_{t-1})\}}(x_t) \end{aligned}$$

for $t > 1$. That is, LININT linearly interpolates between previously seen points, and extrapolates using the value of the hidden function at the nearest previously seen element of the domain. Note that before each trial t, LININT can be thought of as formulating the hypothesis $f_{\{(x_1,y_1),...,(x_{t-1},y_{t-1})\}}$.

Theorem 21:

$$L_2(LININT, \mathcal{F}_2) \leq 1.$$

Proof: Choose a target function $f \in \mathcal{F}_2$ and a sequence $x_1, x_2, ...$ of elements of $[0, 1]$. Assume without loss of generality that the x_t's are distinct.

By Lemma 20, we have that the action of the algorithm's hypothesis increases by at least $(\lambda_t - f(x_t))^2$ on each trial $t > 1$.

Since the function hypothesized after trial 1 is constant, and therefore has action 0, and since, by Lemma 19, the action of LININT's hypothesis is always at most that of the target function, which in turn is at most 1, we may conclude that $\sum_{t>1}(\lambda_t - f(x_t))^2 \leq 1$. □

Note that $||f'||_2 \leq 1$ exactly when $J[f] \leq 1$, and therefore that $\mathcal{F}_2$ can also be thought of as the set of functions whose action is at most 1. The following lemma concerning the function of minimum action subject to certain constraints is well known, and can be proved fairly easily, for instance, through application of an elementary result from the Calculus of Variations (c.f., [Leitmann, 1981, Theorem 2.2][2]).

Lemma 19: *Choose $m \in \mathbf{N}$. Let $(u_1, v_1), ..., (u_m, v_m)$ be a sample. Let $S = \{(u_i, v_i) : 1 \leq i \leq m\}$. If f is a well-behaved function consistent with $(u_1, v_1), ..., (u_m, v_m)$, then*

$$J[f] \geq J[f_S].$$

The following lemma concerns the change in the action of f_S when we add an example to S.

Lemma 20: *Choose $m \in \mathbf{N}$. Let $(u_1, v_1), ..., (u_m, v_m)$ be a sample with $0 \leq u_1 < u_2 < \cdots < u_m \leq 1$. Let $S = \{(u_i, v_i) : 1 \leq i \leq m\}$. Choose an example $(x, y) \in [0, 1] \times \mathbf{R}$. Then*

$$\begin{aligned} J[f_{S \cup \{(x,y)\}}] &\geq J[f_S] + \frac{(y - f_S(x))^2}{\min_i |x - u_i|} \\ &\geq J[f_S] + (y - f_S(x))^2. \end{aligned}$$

Proof: The lemma is trivial if $x < u_1$ or $x > u_m$, and if there is a j for which $x = u_j$. Assume that there is a j such that $u_j < x < u_{j+1}$.

If $a = u_{j+1} - u_j, b = f(u_{j+1}) - f(u_j), c = x_t - u_j$, and $e = (f_S(x_t) - f(x_t)) = (\lambda_t - f(x_t))$ (see Figure 3.1), we can easily see that

$$\begin{aligned} &J[f_{S \cup \{(x_t, f(x_t))\}}] - J[f_S] \\ &\quad = [\tfrac{(\frac{bc}{a}+e)^2}{c} + \tfrac{(b-(\frac{bc}{a}+e))^2}{a-c}] - \tfrac{b^2}{a} \\ &\quad = \tfrac{ae^2}{c(a-c)} \\ &\quad = \tfrac{ae^2}{\min\{c,a-c\}\max\{c,a-c\}} \\ &\quad \geq \tfrac{e^2}{\min\{c,a-c\}}, \end{aligned}$$

completing the proof. □

[2]For those familiar with the Calculus of Variations, the Euler-Lagrange equation in this case is $f''(x) = 0$.

Theorem 18: *If* $p \in \mathbf{R}, p \geq 1$, $LC_p(\mathcal{F}_1) = \infty$.

Proof: The class $\mathcal{F}_1$ includes all continuous twice differentiable increasing functions with $f(0) = 0$ and $f(1) = 1$, since for such functions,

$$\int_0^1 |f'(x)|dx = \int_0^1 f'(x)dx = f(1) - f(0) = 1.$$

The adversary picks $x_1 = 1/2$ and then chooses $f(x_1) = 0$ or $f(x_1) = 1$, whichever gives greater error. Suppose $f(x_1) = 1$. Then the adversary picks $x_2 = 1/4$, and continues the same scheme. If $f(x_1) = 0$, the adversary picks $x_2 = 3/4$ and repeats, et cetera. At each trial the loss is at least $1/2^p$ and there are infinitely many trials. □

3.3 Some positive results

In this section we prove that a very simple algorithm performs optimally with respect to sums of squared errors when the hidden function is in $\mathcal{F}_2$, establishing an alternative proof that $\mathrm{LC}_2(\mathcal{F}_2) = 1$. Loosely speaking, this result implies that the assumption that the average value of the square of the target function's derivative is at most 1 is strong enough for an algorithm to obtain finite worst case bounds on its cumulative squared error. We showed in Section 3.2 that $\mathrm{LC}_2(\mathcal{F}_1) = \infty$.

Suppose $S = \{(u_i, v_i) : 1 \leq i \leq m\}$ is a finite subset of $[0,1] \times \mathbf{R}$ such that

$$u_1 < u_2 < \cdots < u_m.$$

Define $f_S : [0,1] \to \mathbf{R}$ as follows: for all x, $f_\emptyset(x) = 0$, and

$$f_S(x) = \begin{cases} v_1 & \text{if } x \leq u_1 \\ v_i + \frac{(x-u_i)(v_{i+1}-v_i)}{u_{i+1}-u_i} & \text{if } x \in (u_i, u_{i+1}] \\ v_m & \text{if } x > u_m \end{cases}$$

if $|S| \geq 1$.

For $f : [0,1] \to \mathbf{R}$, define the *action* of f, denoted by $J[f]$, to be

$$J[f] = \int_0^1 f'(x)^2 dx. \tag{3.4}$$

By iterating (3.2), concentrating on the second part, we get

$$\mathrm{LC}_1(\mathcal{G}_{a,b}) \geq \frac{jb}{2} + \sum_{i=1}^{j} \mathrm{LC}_1(\mathcal{G}_{a/2^i,0}).$$

Applying Lemma 13, we get

$$\begin{aligned} \mathrm{LC}_1(\mathcal{G}_{a,b}) &\geq \frac{jb}{2} + \left(\sum_{i=1}^{j} a/2^i\right) \mathrm{LC}_1(\mathcal{G}_{1,0}) \\ &= \frac{jb}{2} + (a-b)\mathrm{LC}_1(\mathcal{G}_{1,0}). \end{aligned}$$

This completes the proof. □

Now we are ready to prove the infinite lower bound on $\mathrm{LC}_1(\mathcal{F}_\infty)$.

Theorem 16: *$LC_1(\mathcal{F}_\infty) = \infty$.*

Proof: We will show that even for $\mathcal{G}_{1,0} \subseteq \mathcal{F}_\infty$, $\mathrm{LC}_1(\mathcal{G}_{1,0}) = \infty$.

The adversary chooses $b = 2^{-(j+1)}$ for some $j \in \mathbf{N}$. The adversary then queries the point $x_1 = 1/2$ and chooses $y_1 = b$ or $y_1 = -b$, whichever gives greater error (easily maintaining consistency with a function in $\mathcal{G}_{1,0}$). Then, by Lemma 14, we get two subproblems of $\mathcal{G}_{1/2,b}$. So

$$\begin{aligned} &\mathrm{LC}_1(\mathcal{G}_{1,0}) \\ &\quad \geq b + 2\mathrm{LC}_1(\mathcal{G}_{\frac{1}{2},b}) \\ &\quad \geq b + 2[j\tfrac{b}{2} + (\tfrac{1}{2} - b)\mathrm{LC}_1(\mathcal{G}_{1,0})] \qquad \text{(Lemma 15)} \\ &\quad \geq b + jb + (1-2b)\mathrm{LC}_1(\mathcal{G}_{1,0}). \end{aligned}$$

We can now solve this for $\mathrm{LC}_1(\mathcal{G}_{1,0})$ to get

$$\mathrm{LC}_1(\mathcal{G}_{1,0}) \geq (j+1)/2. \tag{3.3}$$

Since $\mathrm{LC}_1(\mathcal{F}_\infty) \geq \mathrm{LC}_1(\mathcal{G}_{1,0})$ and j was chosen arbitrarily, $\mathrm{LC}_1(\mathcal{F}_\infty) = \infty$. □

As discussed earlier, since $\mathcal{F}_\infty \subseteq \mathcal{F}_q$, $q \geq 1$, this theorem has the following corollary.

Corollary 17: *$LC_1(\mathcal{F}_q) = \infty$ for all $q \geq 1$.*

We may fairly easily see that the assumption that the average value of the (absolute) slope is at most one is not strong enough for practically any positive results in our model.

We begin by showing that $\mathrm{LC}_1(\mathcal{F}_\infty) = \infty$. In contrast, we will show in Section 3.3 that $\mathrm{LC}_2(\mathcal{F}_\infty) = 1$. In our analysis, it will be convenient to consider classes of functions defined on $[0, a]$ for $a > 0$, constrained by the values of the functions at 0 and a.

For $a, b \in [0, 1]$, define $\mathcal{G}_{a,b}$ to be the class of well-behaved functions g defined on $[0, a]$ for which $g(0) = 0$ and $g(a) = b$, with the further restriction that $g'(x) \leq 1$ for all x on which g' is defined.

The following lemmas may be easily verified, e.g., by using reductions between real-valued learning problems [Littlestone *et al.*, 1991] to scale, translate and reflect appropriately.

Lemma 13: *For any $a, c > 0$, $LC_1(\mathcal{G}_{ca,0}) = cLC_1(\mathcal{G}_{a,0})$.*

Lemma 14: *Choose $a, b, c, d \in \mathbf{R}$. Let $\mathcal{H}$ be the class of well-behaved functions f from $[a, b]$ to $\mathbf{R}$ for which $f(a) = c$ and $f(b) = d$, which also have the property that $f'(x) \leq 1$ wherever f' is defined. Then*

$$LC_1(\mathcal{H}) = LC_1(\mathcal{G}_{|b-a|,|c-d|}).$$

We use these lemmas in the following, in which $\mathrm{LC}_1(\mathcal{G}_{a,b})$ is bounded below by a suitable function of $\mathrm{LC}_1(\mathcal{G}_{1,0})$.

Lemma 15: *For $j \in \mathbf{N}$ and $b = 2^{-j}a$,*

$$LC_1(\mathcal{G}_{a,b}) \geq \frac{jb}{2} + (a - b)LC_1(\mathcal{G}_{1,0}). \tag{3.1}$$

Proof: First, note that if $0 \leq b \leq a/2$ then

$$\mathrm{LC}_1(\mathcal{G}_{a,b}) \geq \frac{b}{2} + \mathrm{LC}_1(\mathcal{G}_{\frac{a}{2},0}) + \mathrm{LC}_1(\mathcal{G}_{\frac{a}{2},b}), \tag{3.2}$$

since the adversary may query $a/2$ and answer with whichever of b or 0 gives greater error, while maintaining consistency with a function in $\mathcal{G}_{a,b}$, namely the function which linearly interpolates. In either case there is an immediate error of at least $b/2$ and two subproblems $\mathcal{G}_{a/2,0}$ and $\mathcal{G}_{a/2,b}$ (because of Lemma 14), which the adversary may attack separately.

in the bound. Our results may also be trivially generalized to functions whose range is vector-valued, by treating each component of the predictions and true values separately. We have stated the results in their present form to facilitate presentation of lower bounds, as well as to cut down on unnecessary notation, as we feel that the essence of the problems is captured in the simple cases.

Faber and Mycielski [Faber and Mycielski, 1991] proved, using a different algorithm, that $\mathrm{LC}_2(\mathcal{F}_2) \leq 1$. This result amounts to a special case of a beautiful theorem about learning linear functionals defined on Hilbert spaces using a generalization of the Widrow-Hoff algorithm [Widrow and Hoff, 1960], and their paper contains numerous other applications of their Hilbert space results. Nonetheless, we feel it is interesting that even the very simple linear interpolation algorithm is optimal for $\mathcal{F}_2$ with respect to sums of squared errors.

Many statisticians, and, more recently, computational learning theorists (c.f., [Hardle, 1991] [Barron, 1991] [Haussler, 1989a]) have studied the induction of classes of functions obtained through smoothness constraints. The spirit of their work differs from ours in several ways. First, their theorems usually concern functions of potentially many real variables, where ours, at present, apply only to functions of a single real variable. On the other hand, the previous work usually involves use of probabilistic assumptions on the generation of the x_t's, for instance that they are drawn independently from a fixed distribution on whatever domain, whereas our results do not use such assumptions. These assumptions have enabled researchers to prove bounds on the expected "loss" on a particular trial. In worst-case models such as that considered here, such "instantaneous" bounds are clearly impossible (c.f., [Littlestone, 1989b]). Finally, in many cases, we are able to obtain upper and lower bounds that match, including constants, which is often not the case for the previously studied problems.

3.2 Some negative results

In this section, we describe several settings in which no algorithm can acheive any finite bound on the cumulative loss.

the hidden function at the nearest previously seen point.[1] We show that the worst-case sum of squared errors made by this algorithm while learning $\mathcal{F}_2$ is 1. A trivial lower bound establishes the fact that this algorithm is optimal for $\mathcal{F}_2$ with respect to the worst-case sum of squared errors, and therefore that $\mathrm{LC}_2(\mathcal{F}_2) = 1$.

Since, as is easily verified, the 1-norm of a function is at most its 2-norm which is in turn at most its ∞-norm, we have that $\mathcal{F}_\infty \subseteq \mathcal{F}_2 \subseteq \mathcal{F}_1$. Combining the first inequality with the positive result above implies that $\mathrm{LC}_2(\mathcal{F}_\infty) \leq 1$. Again, a trivial lower bound shows that this is the best possible, and therefore that $\mathrm{LC}_2(\mathcal{F}_\infty) = 1$. Similarly, it follows from our main negative result that $\mathrm{LC}_1(\mathcal{F}_1) \geq \mathrm{LC}_1(\mathcal{F}_2) \geq \mathrm{LC}_1(\mathcal{F}_\infty) = \infty$. A simple argument establishes that $\mathrm{LC}_p(\mathcal{F}_1) = \infty$ for all $p \geq 1$.

We next show that $\mathrm{LC}_p(\mathcal{F}_\infty) \leq 1 + 1/(2^p - 2)$ for $p > 1$. Combining this with the aforementioned results about $\mathcal{F}_\infty$, we may conclude that $\mathrm{LC}_p(\mathcal{F}_\infty) < \infty$ exactly when $p > 1$. For this upper bound, we analyze the algorithm which simply predicts with the value of the hidden function at the nearest previously seen element of the domain, which, though intuitively worse than the "linear interpolation" algorithm, is easier to analyze. We also prove that if $p > 1$, we have $\mathrm{LC}_p(\mathcal{F}_2) \leq 2 + 1/(2^p - 2)$, which implies that $\mathrm{LC}_p(\mathcal{F}_2)$ is also finite exactly when $p > 1$.

Finally, we describe some preliminary results concerning bounded length sequences of trials, showing that the sum of (unsquared) errors made by either of the above algorithms learning $\mathcal{F}_\infty$ and $\mathcal{F}_2$ respectively on trial sequences of length m is at most $e(1 + (\log m)/2)$.

Our analyses can trivially be extended to classes of functions defined on an arbitrary interval, and to classes formed through arbitrary bounds on the various norms of the derivatives. Furthermore, the algorithms we describe do not make use of knowledge of the endpoints of the interval, or of knowledge of how steep the target function tends to be. Therefore, we may even view our upper bounds as applying to arbitrary well-behaved functions of the entire real line, where the maximum magnitude of an element of the domain encountered in a sequence of trials, as well as the steepness of the target function, appears

[1] On the very first trial, it predicts arbitrarily, say with 0.

3. The Learning Complexity of Smooth Functions of a Single Variable

3.1 Introduction

In this chapter, we will consider learning functions of a single real variable. We will further assume that the domain is simply $[0, 1]$, although we will see later that this restriction is only for convenience, and meaningful results can be obtained without it. We will also limit our attention to continuous functions that are piecewise twice differentiable (i.e., twice differentiable except on a finite set). Let's call such functions *well-behaved*.

We wish to model the intuition that, for many functions encountered in practice, similar inputs tend to yield similar outputs. Toward this end, for $q \in \{1, 2, \infty\}$, we will study the class $\mathcal{F}_q$ of well-behaved functions whose first derivatives have q-norm at most 1. Recall that, for $1 \leq q < \infty$, the q-norm of a function f defined on $[0, 1]$ is defined to be

$$\left(\int_0^1 |f(x)|^q dx \right)^{1/q},$$

and that the infinity norm of f is the limit, as q approaches infinity, of its q-norm. The infinity norm roughly corresponds to the maximum value of $|f(x)|$, and the one-norm, to the average, while the two-norm lies somewhere in between. Thus, $\mathcal{F}_\infty$ roughly corresponds to the class of functions that are never very steep, and $\mathcal{F}_1$ to the class of functions that are not very steep on average.

In this chapter, we determine the value of $\mathrm{LC}_p(\mathcal{F}_q)$ for each $(p, q) \in \{1, 2\} \times \{1, 2, \infty\}$.

Our main negative result is that $\mathrm{LC}_1(\mathcal{F}_\infty) = \infty$. This result, loosely speaking, says that even the assumption that the hidden function *never* has slope greater than one is not sufficiently strong to enable an algorithm to obtain any finite bound on the sum of the absolute values of the differences between predictions and true values.

Our main positive result concerns the algorithm which at each trial linearly interpolates between previously seen function values, and extrapolates by predicting with the value of

whether a similar algorithm is optimal for learning the class containing all linear functions composed with the standard sigmoid function $(1/(1+e^{-x}))$. One can trivially obtain bounds from our results, but they appear to be suboptimal.

2.5 Discussion

Linear functions are widely used. We expect that our algorithm may become a standard submodule for learning more complicated functions or for learning linear combinations of previously learned functions.

The fact that our algorithm must know a bound on the sum of the absolute values of coefficients of the target function might make it appear somewhat unattractive to practitioners. However, this problem may be circumvented by application of the Weighted Majority algorithm [Littlestone and Warmuth, 1989] to an pool consisting of algorithms that assume various upper bounds on the size of the hidden coefficient vectors. Nevertheless, to simplify application of our techniques to real-world problems, it would be useful to have a variant of our algorithm for which we can directly obtain bounds similar to our present bounds without knowing anything about the hidden coefficients.

We also are interested in improving our lower bounds. Is it possible that similar lower bounds hold even when the algorithm has more information about the hidden coefficients, or even about the upcoming sequence of examples?

We are also investigating the case in which the coefficient vector changes gradually over time, corresponding to a case in which some linear combination of the economists is close to the actual GNP for a certain period, and then in later periods other linear combinations do well. The algorithm is to "track" the best linear combination with some additional cost that grows as a function of how much the coefficient vector changes over time. This would generalize the methods of [Littlestone and Warmuth, 1989] with which one could track the best single economist.

In addition, it would be interesting to find algorithms which are optimal with respect to other natural loss functions, in particular, $|\lambda_t - \rho_t|$. Recently, Bernstein [Bernstein, 1992] has proved a lower bound for this problem whose dependence on n is $\sqrt{n}$, and has described an algorithm whose worst-case sum of absolute errors is $O(\sqrt{n \log n})$.

Finally, since our algorithms have a similar flavor to the linear threshold algorithms of [Littlestone, 1988] [Littlestone and Warmuth, 1989] [Littlestone, 1989b], one might ask

For the first stage, which consists of $n-1$ examples, the tth instance is given by

$$x_i^{(t)} = \begin{cases} \text{M} & \text{if } i = t \\ 0 & \text{otherwise} \end{cases}$$

and the tth response is always 0. Note that if for each t, $v^{(t)} \in \mathbf{R}^n$ is defined by

$$v_i^{(t)} = \begin{cases} \text{c} & \text{if } i = t \\ 0 & \text{otherwise} \end{cases}$$

then for each $t \leq n-1$, $v^{(t)}$ is consistent with the first $t-1$ examples, and thus minimizes the observed loss on these examples. Yet if $\lambda^{(t)}$ is the prediction made using $v^{(t)}$, then for each t, $\lambda^{(t)} = cM$, thus

$$\sum_{t=1}^{n-1} (\lambda^{(t)} - \rho^{(t)})^2 = (cM)^2(n-1). \tag{2.9}$$

Note that $v^{(n)}$ is consistent with all the examples of the first stage.

The second stage is virtually identical to the second stage of Theorem 9, replacing $(1/2, ..., 1/2)$ with $(0, ..., 0)$, and responding with whichever of $-cM$ and cM is farthest from the algorithm's prediction. One can easily see that, as in Theorem 9, the algorithm can be forced to have total loss of N in the second stage. Combining this with (2.9) yields the desired result. □

The Widrow-Hoff algorithm predicts using an unnormalized weight vector which is updated after each trial, i.e. the algorithm's prediction on trial t is $\vec{w}_t \cdot \vec{x}_t$, where each $\vec{w}_t = (w_{t,1}, ..., w_{t,n})$ is defined as follows. The initial weight vector $\vec{w}_1$ is the zero vector, and subsequent weight vectors are obtained from the examples according to the following rule:

$$\vec{w}_{t+1} = \vec{w}_t + (\rho_t - \lambda_t)\frac{\vec{x}_t}{\vec{x}_t \cdot \vec{x}_t}.$$

The following lower bound has been proved for this algorithm.

Theorem 12 ([Cesa-Bianchi *et al.*, 1991]): *Let W be the Widrow-Hoff algorithm. For each $n \in \mathbf{N}$, $M, c, N > 0$,*

$$L_2(W, \text{LINEAR}(n, M, c), N) \geq (cM)^2 n + N.$$

Recall that the total loss of our algorithm was $O((cM)^2 \log n + N)$.

computing weighted averages, which is quite surprising. Classes of weighted-average functions whose weights have high entropy (which requires many non-zero weights) are easier to learn. This is in contrast to the case of learning boolean functions, such as boolean linear-threshold functions, where in general (for classes closed under permutation of the attributes) learning gets harder as the number of relevant variables increases [Littlestone, 1988] [Littlestone and Warmuth, 1989] [Littlestone, 1989b] [Blum *et al.*, 1991].[6] (Some of the upper bounds of [Littlestone, 1989b] depend on a product of two factors, one of which shows the same decreasing dependence on entropy observed here; that decrease is typically dwarfed by an increase in the other factor as the number of relevant variables increases.)

The following is a straightforward extension of the previous theorem. Its proof is therefore omitted.

Corollary 10: *We have*

$$\begin{aligned} LC_2(\mathrm{WA}(n, M, \kappa), N) &\in \Omega(M^2(\ln n - \kappa) + N) \\ LC_2(\mathrm{LINEAR}(n, M, c), N) &\in \Omega((cM)^2 \ln n + N). \end{aligned}$$

By a least squares algorithm, we mean any algorithm which hypothesizes a linear function at each trial that minimizes the sum of the squared errors on previous examples. Next, we show that a least squares algorithm can have total loss which depends linearly on the number of variables n.

Theorem 11: *For each $n \in \mathbf{N}$, $M, c, N > 0$, there exists a least squares algorithm B such that*

$$L_2(B, \mathrm{LINEAR}(n, M, c), N) \geq (cM)^2(n-1) + N.$$

Proof: Choose $n \in \mathbf{N}$, $M, c, N > 0$. As before, after the adversary gives a first example of $((0, ..., 0), 0)$, the adversary strategy is broken into two stages. In the first stage, the adversary maintains consistency with some element of $\mathrm{LINEAR}(n, M, c)$, and in the second stage, the adversary greedily expends a "noise budget."

[6] For certain especially simple classes mistake bounds can again drop as the number of relevant variables becomes a significant fraction of all of the variables.

In the second stage, which consists of $\lfloor 4N \rfloor + 1$ trials, each instance is $(1/2, 1/2, ..., 1/2)$, and for the first $\lfloor 4N \rfloor$ trials the adversary simply responds with whichever of 0 or 1 is further from the algorithm's prediction. On the last trial, if the algorithm's prediction is no more than $1/2$, the adversary responds with $1/2 + (1/2)\sqrt{4N - \lfloor 4N \rfloor}$, otherwise he responds with $1/2 - (1/2)\sqrt{4N - \lfloor 4N \rfloor}$.

Let $m = l - k + \lfloor 4N \rfloor + 1$ be the total number of trials of the adversary. Since the fact that $\vec{\mu} \cdot (1/2, ..., 1/2)$ must equal $1/2$ implies that for each $t, l - k < t < m$,

$$(\vec{\mu} \cdot \vec{x}^{(t)} - \rho^{(t)})^2 = 1/4,$$

we have

$$\sum_{t<m} (\rho^{(t)} - \vec{\mu} \cdot \vec{x}^{(t)})^2 = \frac{\lfloor 4N \rfloor}{4}.$$

Also,

$$(\rho^{(m)} - \vec{\mu} \cdot \vec{x}^{(t)})^2 = \frac{4N - \lfloor 4N \rfloor}{4},$$

so

$$\sum_{t} (\rho^{(t)} - \vec{\mu} \cdot \vec{x}^{(t)})^2 = \frac{4N}{4} = N.$$

Also, the loss on each trial t of phase two is at least $(\vec{\mu} \cdot \vec{x}^{(t)} - \rho^{(t)})^2$, thus the total loss of stage two is at least N.

Combining this with (2.8) yields the desired result. □

Note that this argument proves a stronger result than that stated in the theorem, since all of the instances of the sequence of examples, as well as the entropy of the hidden coefficient vector and the amount of noise, may be given to the algorithm before the first prediction is made and adversary can then choose the responses of each example so that the loss is maximized.

Note also that in the case that $\kappa = 0$, the adversary uses only functions with just one nonzero coefficient. This, combined with Theorem 7, implies that the inherent complexity of the problem of learning functions which simply output a selected component is the same (at least to within a constant factor) as that of learning the class of all functions

Using similar techniques, we can easily prove similar theorems for classes formed by linear combinations of functions taken from some fixed finite set, e.g. for bounded degree polynomials.

2.4 Lower bounds

We begin by proving a lower bound on $\mathrm{LC}_2(\mathrm{WA}(n,1,\kappa),N)$. Our more general lower bounds can be derived from this initial result. For the proof, we will need the following notation. For $u,v \in \mathbf{N}, v \leq \log u + 1$, let $\mathrm{bit}(u,v)$ be the vth least significant bit of the binary representation of u (e.g., $\mathrm{bit}(6,1)=0, \mathrm{bit}(6,2)=1, \mathrm{bit}(6,3)=1$).

Theorem 9: *We have*

$$LC_2(\mathrm{WA}(n,1,\kappa),N) \geq \frac{(\ln n - \kappa)}{4\ln 2} + N - \frac{1}{2}.$$

Proof: Let $l = \lfloor \log n \rfloor, k = \lceil \kappa/(\ln 2) \rceil$. Consider an adversary which adaptively constructs a sequence of examples as follows. The adversary's first example is $((0,...,0),0)$. Afterwards, the adversary operates in two stages. In the first stage, the adversary maintains consistency with some function in $\mathrm{WA}(n,1,k) \subseteq \mathrm{WA}(n,1,\kappa)$. In the second stage, the adversary greedily uses up its "noise budget."

The instances $\vec{x}^{(1)},...,\vec{x}^{(l-k)}$ of the first stage are constructed as follows: $x_i^{(t)} = 1$ if $\mathrm{bit}(i,t) = 1$ and $i \leq 2^l$, otherwise $x_i^{(t)} = 0$. The adversary responds with 1 if the algorithm's prediction is no more than $1/2$, otherwise the adversary responds with 0. Thus the loss of the algorithm on each trial of stage one is at least $1/4$.

Define $\vec{\mu}$ as follows: if $i \leq 2^l$ and for each $t \leq l-k$, $\mathrm{bit}(i,t) = \rho^{(t)}$, then let $\mu_i = 2^{-k}$, and otherwise, let $\mu_i = 0$. Since the number of l bit vectors "satisfying" a $(l-k)$-bit mask is 2^k, $||\vec{\mu}||_1 = 1$. Also, by construction, the linear function induced by $\vec{\mu}$ is consistent with the examples of the first phase. Trivially, $H(\vec{\mu}) = k\ln 2 \geq \kappa$. Since the first phase consists of $l-k$ trials, the total loss of the first phase is at least

$$\frac{1}{4}(\lfloor \ln n/(\ln 2) \rfloor - \lceil \kappa/(\ln 2) \rceil). \tag{2.8}$$

bounds given later in this chapter show that tuning δ can only yield an improvement of a constant factor over the choice $\delta = 1/\sqrt{2}$.

2.2.5 Noise tolerance

Note that the smallest we can make the constant on the "noise term" (at the expense of the term depending on n and $H(\vec{\mu})$) by increasing δ is 4. However, our analysis is somewhat loose, which leaves open the possibility that our algorithm's loss (or that of a related algorithm) is bounded by $k(\ln n - H(\vec{\mu})) + N(\vec{\mu})$ for some constant k.

2.3 More general linear functions

In this section, we describe more general learning results, obtained by applying the reductions between real-valued learning problems described in Appendix C. All proofs of this section are relatively straightforward, and may be found in that appendix.

For each $n \in \mathbf{N}, M, \kappa, c > 0$ we will need the following definitions. Let $\mathrm{WA}(n, M, \kappa)$ be the set of $f : [0, M]^n \to [0, M]$ such that there exists $\vec{\mu} \in [0, 1]^n, ||\vec{\mu}||_1 = 1$ whose entropy is at least κ such that $f(\vec{x}) = \vec{\mu} \cdot \vec{x}$ for all $\vec{x}$. Let $\mathrm{LINEAR}(n, M, c)$ be the set of linear functions defined on $[0, M]^n$ such that the sum of the absolute values of their coefficients is at most c. Since the entropy is only defined for non-negative coefficients summing to 1, we omit the entropy parameter from LINEAR.

This section's theorem shows that our algorithm may be modified to obtain excellent algorithms for the classes defined above.

Theorem 8:

$$LC_2(\mathrm{WA}(n, M, \kappa), N) \leq M^2 LC_2(\mathrm{WA}(n, 1, \kappa), N/M^2) \in O(M^2(\ln n - \kappa) + N)$$

$$LC_2(\mathrm{LINEAR}(n, M, c), N) \leq (2cM)^2 LC_2(\mathrm{WA}(2n + 1, 1, 0), N/(2cM)^2) \in O((cM)^2 \ln n + N)$$

Proof: In Appendix C. □

Note that $\exp(\Delta_t) = \sum_{i=1}^{n} v_{t,i}\beta_t^{x'_{t,i}-\rho'_t}$, and therefore that

$$\frac{\partial \exp(\Delta_t)}{\partial \beta_t} = 0 \text{ iff } \rho_t = \vec{v}_{t+1} \cdot \vec{x}_t \text{ and}$$
$$\frac{\partial^2 \exp(\Delta_t)}{\partial^2 \beta_t} \geq 0 \text{ when} \frac{\partial \exp(\Delta_t)}{\partial \beta_t} = 0 \text{ and } \beta_t \geq 0.$$

Thus $\exp(\Delta_t)$, and therefore Δ_t, has exactly one minimum when $\beta_t \in [0, \infty]$. Denote the β_t at the minimum as $\beta_{t,opt}$. Now if we updated with $\beta_{t,opt}$ and fed $\vec{x}'_t$ to A_δ after the update was made, the algorithm would predict ρ_t. Thus with the optimum choice for β_t the algorithm is in some sense "corrective."

Since we have determined the choice for β_t which gives the best bound when $\rho_t = \vec{\mu} \cdot \vec{x}_t$, why not use it? First, we know no closed form for $\beta_{t,opt}$. We can use a number of heuristics for approximating $\beta_{t,opt}$ such as gradient descent, Newton's method or binary search. Another choice is to iterate the update of A_δ a number of times with the same instance $\vec{x}_t$.

However, even if the computational cost of approximating $\beta_{t,opt}$ is not a deterrent, there is a second reason for not choosing a β_t that is too close to $\beta_{t,opt}$. This is illustrated with the following example. Assume there is a long sequence of examples consistent with $\vec{\mu} = (1/2, 1/2)$ except that the first example $((1,0),1)$ is noisy. In this case, in order to be consistent, we must hypothesize $\vec{v}_2 = (1,0)$, effectively choosing $\beta_1 = \infty$. Now all future updates cannot correct the second component of the weight vector of $\vec{v}_2$, leading to an unbounded loss on future examples consistent with $(1/2, 1/2)$.

So in case of noise it is advantageous to choose β_t not too close to $\beta_{t,opt}$ and instead make a less drastic update.

2.2.4 Tuning δ

If one has a prior idea of $N(\vec{\mu})$ ahead of time, one can tune δ to optimize the first bound of the preceding theorem. In fact, if N is known, one may calculate the best δ using calculus. However, the resulting expression, as a function of N and n, is hideous, and lower

2.2.1 Choosing an initial weight vector

If we choose $\vec{v}_1$ to be something other than $(1/n, ..., 1/n)$, reflecting some prior bias on which weighted combination of the experts predicts well, then the bounds in the previous theorem hold if we replace "$\ln n - H(\vec{\mu})$" by "$I(\vec{\mu}||\vec{v}_1)$". Thus, our algorithm can take advantage of increasingly accurate prior beliefs.

2.2.2 Trading between fit and entropy

There is a curious trade off between $N(\vec{\mu})$ and $H(\vec{\mu})$ in the upper bound

$$\mathrm{L}_2(A_{1/\sqrt{2}}, S) \leq 5.83(\ln n + \min_{\vec{\mu}}(N(\vec{\mu}) - H(\vec{\mu}))).$$

For example, assume the algorithm receives a single example $((1, 0, \cdots, 0), 1)$. Since we require that $\vec{\mu} \in [0, 1]^n$ and $||\vec{\mu}||_1 = 1$, only $\vec{\mu}_1 = (1, 0, \cdots, 0)$ is consistent. The upper bound for $\vec{\mu} = \vec{\mu}_1$ is $5.83 \ln n$, since $N(\vec{\mu}_1) = H(\vec{\mu}_1) = 0$. However for $\vec{\mu} = (1/n, 1/n, \cdots, 1/n)$ the bound is 5.83, so the minimum in the loss bound is not achieved at the consistent vector $\vec{\mu}_1$.

2.2.3 Choosing the base of the exponent in our update

How did we come up with our choice of $\beta_t = \frac{\rho_t+\delta}{\lambda_t+\delta}\frac{1-\lambda_t+\delta}{1-\rho_t+\delta}$ as the base of the exponent in our update for the algorithm A_δ? Consider the upper bound for Δ_t given by the Inequality (2.5) for the case when $\rho_t = \vec{\mu} \cdot \vec{x}_t$:

$$\Delta_t \leq \ln(1 + (\beta_t - 1)\lambda'_t) - \rho'_t \ln \beta_t.$$

Our above choice for β_t is obtained by minimizing this upper bound for Δ_t, i.e. we maximize our bounds on the decrease of $I(\vec{\mu}||\vec{v}_t)$ caused by the update in trial t.

However, there are better choices for β_t for the case when $\rho_t = \vec{\mu} \cdot \vec{x}_t$. From (2.4) we get

$$\Delta_t = \ln(\sum_{i=1}^{n} v_{t,i}\beta_t^{x'_{t,i}}) - \rho'_t \ln \beta_t.$$

where the minimum is over all $\vec{\mu} \in [0,1]^n$ *with* $||\vec{\mu}||_1 = 1$ *and for each such* $\vec{\mu}$, $N(\vec{\mu})$ *is defined to be* $\sum_{t=1}^{m}(\vec{\mu} \cdot \vec{x}_t - \rho_t)^2$. *In particular,*

$$L_2(A_{1/\sqrt{2}}, S) \leq 5.83(\ln n + \min_{\vec{\mu}}(N(\vec{\mu}) - H(\vec{\mu}))).$$

Further, for any sequence $S = \langle(\vec{x}_t, \rho_t)\rangle_{1 \leq t \leq m}$ *of* m *examples in* $[0,1]^n \times [0,1]$ *for which there exists* $\vec{\mu} \in [0,1]^n, ||\vec{\mu}||_1 = 1$ *such that for all* $t, 1 \leq t \leq m$, $\vec{\mu} \cdot \vec{x}_t = \rho_t$, *we have*

$$L_2(A_0, S) \leq \frac{(\ln n - H(\vec{\mu}))}{2}.$$

Proof: Assume first that $\delta > 0$. Since $I(\vec{\mu}||\vec{v}_1) = \ln n - H(\vec{\mu})$ and and $I(\vec{\mu}||v_{m+1}) \geq 0$,

$$\sum_{t=1}^{m} \Delta_t = I(\vec{\mu}||v_{m+1}) - I(\vec{\mu}||\vec{v}_1) \geq -\ln n + H(\vec{\mu}).$$

Thus using (2.2), we get:

$$\sum_{t=1}^{m} - \frac{2}{(1+2\delta)^2}(\lambda_t - \rho_t)^2 + \frac{|\rho_t - \vec{\mu} \cdot \vec{x}_t||\rho_t - \lambda_t|}{\delta(1+\delta)} \geq -\ln n + H(\vec{\mu}).$$

Rewrite the second inequality

$$\sum_{t=1}^{m} - \left(\frac{\sqrt{2}|\lambda_t - \rho_t|}{1+2\delta}\right)^2 + \left(\frac{\sqrt{2}|\rho_t - \lambda_t|}{1+2\delta}\right)\left(\frac{(1+2\delta)|\rho_t - \vec{\mu} \cdot \vec{x}_t|}{\sqrt{2}\delta(1+\delta)}\right) \geq -\ln n + H(\vec{\mu})$$

and apply Lemma 3, obtaining

$$\sum_{t=1}^{m} - \frac{1}{(1+2\delta)^2}(\lambda_t - \rho_t)^2 + \frac{(1+2\delta)^2}{4\delta^2(1+\delta)^2}(\rho_t - \vec{\mu} \cdot \vec{x}_t)^2 \geq -\ln n + H(\vec{\mu}).$$

Solving for $\sum_{t=1}^{m}(\lambda_t - \rho_t)^2$ yields the first loss bound of the theorem:

$$\mathrm{L}_2(A_\delta, S) \leq (1+2\delta)^2(\ln n - H(\vec{\mu})) + \frac{(1+2\delta)^4}{4\delta^2(1+\delta)^2}N(\vec{\mu}).$$

For the second bound, observe that when $\delta = 1/\sqrt{2}$,

$$(1+2\delta)^2 = \frac{(1+2\delta)^4}{4\delta^2(1+\delta)^2} \leq 5.83.$$

Finally, the third bound may be easily obtained in a similar manner using (2.3).

This completes the proof. □

$$\begin{aligned}
\Delta_t &= \sum_{i=1}^{n} \mu_i \ln \frac{v_{t,i}}{v_{t+1,i}} \\
&= \sum_{i=1}^{n} \mu_i \left(\lim_{\gamma \to 0} \ln \frac{\sum_{j=1}^{n} v_{t,j} {\beta_\gamma}^{x_{t,j}}}{\beta_\gamma^{x_{t,i}}} \right) \\
&= \lim_{\gamma \to 0} \sum_{i=1}^{n} \mu_i \left(\ln \frac{\sum_{j=1}^{n} v_{t,j} {\beta_\gamma}^{x_{t,j}}}{\beta_\gamma^{x_{t,i}}} \right) \\
&\leq \lim_{\gamma \to 0} -2 \frac{(\lambda_t - \rho_t)^2}{(1+2\gamma)^2} \qquad \text{(Using (2.6))} \\
&= -2(\lambda_t - \rho_t)^2
\end{aligned}$$

establishing (2.3).

Next, assuming again that $\delta > 0$, replacing the second term in (2.6) with its absolute value, we obtain:

$$\Delta_t \leq -2(\rho'_t - \lambda'_t)^2 + \frac{|\rho_t - \vec{\mu} \cdot \vec{x}_t| \; |\ln \beta_\delta|}{1+2\delta}. \tag{2.7}$$

Now, we wish to bound $|\ln \beta|$. First, let us assume that $\lambda_t \leq \rho_t$. Let $z = \rho_t - \lambda_t$. Then

$$\ln \beta = \ln \frac{(\lambda + z + \delta)(1 - \lambda + \delta)}{(\lambda + \delta)(1 - \lambda - z + \delta)}.$$

Applying Lemmas 4 and 5, we get that

$$\ln \beta \leq \frac{(2\delta+1)z}{\delta(1+\delta)} = \frac{(2\delta+1)(\rho_t - \lambda_t)}{\delta(1+\delta)}.$$

By symmetry, when $\rho_t \leq \lambda_t$, if we let $z = \lambda_t - \rho_t$, we obtain

$$\ln \frac{1}{\beta} \leq \frac{(2\delta+1)z}{\delta(1+\delta)} = \frac{(2\delta+1)(\lambda_t - \rho_t)}{\delta(1+\delta)}.$$

Hence

$$|\ln \beta| \leq \frac{(2\delta+1)z}{\delta(1+\delta)} = \frac{(2\delta+1)|\rho_t - \lambda_t|}{\delta(1+\delta)}.$$

Plugging into (2.7) yields the desired result. □

We can apply the previous lemma to obtain the following loss bounds.

Theorem 7: *Choose $n, m \in \mathbf{N}$. Let $S = \langle (\vec{x}_t, \rho_t) \rangle_{1 \leq t \leq m}$ be any sequence of m examples in $[0,1]^n \times [0,1]$. Then for each $\delta > 0$,*

$$L_2(A_\delta, S) \leq \min_{\vec{\mu}} \left((1+2\delta)^2 (\ln n - H(\vec{\mu})) + \frac{(1+2\delta)^4}{4\delta^2(1+\delta)^2} N(\vec{\mu}) \right)$$

coefficient vectors hypothesized by A_δ and $\langle\lambda_t\rangle_{t\in\mathbf{N}}$ be the sequence of A_δ's predictions. Let $\Delta_t = I(\vec{\mu}||\vec{v}_{t+1}) - I(\vec{\mu}||\vec{v}_t)$ and for $z \in \mathbf{R}$ let z' denote $\frac{z+\delta}{1+2\delta}$. Then for all t, if $\delta > 0$,

$$\Delta_t \leq -\frac{2}{(1+2\delta)^2}(\rho_t - \lambda_t)^2 + \frac{|\rho_t - \vec{\mu}\cdot\vec{x}_t||\rho_t - \lambda_t|}{\delta(1+\delta)}. \tag{2.2}$$

If $\delta = 0$ and $\rho_s = \vec{\mu}\cdot\vec{x}_s$ for all $s \leq t$,

$$\Delta_t \leq -2(\lambda_t - \rho_t)^2. \tag{2.3}$$

Proof: Choose t. For each $\delta > 0$, let

$$\beta_\delta = \left(\frac{\rho_t + \delta}{\lambda_t + \delta}\right)\left(\frac{1 - \lambda_t + \delta}{1 - \rho_t + \delta}\right).$$

Assume for the moment that $\delta > 0$. From the definition of Δ_t and from Lemma 2 it follows that

$$\begin{aligned}
\Delta_t &= \sum_{i=1}^n \mu_i \ln \frac{v_{t,i}}{v_{t+1,i}} \\
&= \sum_{i=1}^n \mu_i \ln \frac{\sum_{j=1}^n v_{t,j}\beta_\delta^{\frac{x_{t,j}}{1+2\delta}}}{\beta_\delta^{\frac{x_{t,i}}{1+2\delta}}} \\
&= \sum_{i=1}^n \mu_i \ln \frac{\sum_{j=1}^n v_{t,j}\beta_\delta^{x'_{t,j}}}{\beta_\delta^{x'_{t,i}}} \\
&= \ln(\sum_{i=1}^n v_{t,i}\beta_\delta^{x'_{t,i}}) + \sum_{i=1}^n \mu_i \ln \frac{1}{\beta_\delta^{x'_{t,i}}} \qquad (2.4) \\
&\leq \ln(\sum_{i=1}^n v_{t,i}(1 + (\beta_\delta - 1)x'_{t,i})) - \sum_{i=1}^n \mu_i x'_{t,i} \ln \beta_\delta \\
&= \ln(1 + (\beta_\delta - 1)\lambda'_t) - \frac{\vec{\mu}\cdot\vec{x}_t + \delta}{1+2\delta}\ln\beta_\delta \\
&= \ln(1 + (\beta_\delta - 1)\lambda'_t) - \rho'_t \ln\beta_\delta + \frac{\rho_t - \vec{\mu}\cdot\vec{x}_t}{1+2\delta}\ln\beta_\delta. \qquad (2.5)
\end{aligned}$$

Since β_δ can be written as $\frac{\rho'_t}{\lambda'_t}\frac{1-\lambda'_t}{1-\rho'_t}$ we can rewrite the last expression as

$$-I((\rho'_t, 1-\rho'_t)||(\lambda'_t, 1-\lambda'_t)) + \frac{\rho_t - \vec{\mu}\cdot\vec{x}_t}{1+2\delta}\ln\beta_\delta \leq -2\frac{(\lambda_t - \rho_t)^2}{(1+2\delta)^2} + \frac{\rho_t - \vec{\mu}\cdot\vec{x}_t}{1+2\delta}\ln\beta_\delta, \tag{2.6}$$

applying Lemma 1. Since, if $\delta = 0$ and $\rho_t = \vec{\mu}\cdot\vec{x}$,

$$v_{t+1,i} = \frac{v_{t,i}\left(\frac{(\rho_t+\delta)(1-\lambda_t+\delta)}{(\lambda_t+\delta)(1-\rho_t+\delta)}\right)^{\frac{x_{t,i}}{1+2\delta}}}{\sum_{i=1}^{n} v_{t,i}\left(\frac{(\rho_t+\delta)(1-\lambda_t+\delta)}{(\lambda_t+\delta)(1-\rho_t+\delta)}\right)^{\frac{x_{t,i}}{1+2\delta}}}$$

If $\delta = 0$, the update is

$$v_{t+1,i} = \lim_{\gamma \to 0} \frac{v_{t,i}\left(\frac{(\rho_t+\gamma)(1-\lambda_t+\gamma)}{(\lambda_t+\gamma)(1-\rho_t+\gamma)}\right)^{\frac{x_{t,i}}{1+2\gamma}}}{\sum_{i=1}^{n} v_{t,i}\left(\frac{(\rho_t+\gamma)(1-\lambda_t+\gamma)}{(\lambda_t+\gamma)(1-\rho_t+\gamma)}\right)^{\frac{x_{t,i}}{1+2\gamma}}} \tag{2.1}$$

The algorithm A_0 is only intended for use when there is known to be "no noise,"[5] i.e. when there is a probability vector $\vec{\mu}$ such that for all t, $\rho_t = \vec{\mu} \cdot \vec{x}_t$. In such circumstances, a simple but tedious proof, included in Appendix B establishes that the weights maintained by A_0 are always finite. If $\rho_t < 1$ and $\lambda_t > 0$, it is trivial that the above update preserves the finiteness of the weights, and we may replace the limit above with the simpler update:

$$v_{t+1,i} = \frac{v_{t,i}\left(\frac{\rho_t(1-\lambda_t)}{\lambda_t(1-\rho_t)}\right)^{x_{t,i}}}{\sum_{i=1}^{n} v_{t,i}\left(\frac{\rho_t(1-\lambda_t)}{\lambda_t(1-\rho_t)}\right)^{x_{t,i}}}.$$

In the case that $\rho_t = 1$, Appendix B contains a proof that the update of (2.1) is equivalent to the following

$$v_{t+1,i} = \begin{cases} \frac{v_{t,i}}{\sum_{j:\, x_{t,j}=1} v_{t,j}} & \text{if } x_{t,i} = 1 \\ 0 & \text{otherwise.} \end{cases}$$

When $\lambda_t = 0$, Appendix B contains a proof that $\rho_t = 0$ (again, assuming that there is "no noise"), and therefore trivially, that using A_0, for all i, $v_{t+1,i} = v_{t,i}$.

As in [Littlestone, 1989b] in the case of linear threshold algorithms, we use the relative entropy between the coefficient vector $\vec{\mu}$ of a target function and the coefficient vector $\vec{v}_t$ of the algorithm's hypothesis as a measure of progress. Our key lemma relates the change in this measure of progress on a particular trial to the loss of the algorithm on that trial. Loosely speaking, it says that the algorithm learns a lot when it makes large errors.

Lemma 6: *Choose $\delta \geq 0$ and $n \in \mathbf{N}$. Choose $\vec{\mu} \in [0,1]^n$ such that $||\vec{\mu}||_1 = 1$. Let $\langle(\vec{x}_t, \rho_t)\rangle_{t \in \mathbf{N}}$ be a sequence of examples from $[0,1]^n \times [0,1]$. Let $\langle \vec{v}_t \rangle_{t \in \mathbf{N}}$ be the sequence of*

[5] Even then, it is not recommended for numerical reasons.

Lemma 5: *For all $\delta > 0$, and z such that $0 \leq z \leq 1$,*

$$\ln \frac{(z+\delta)(1+\delta)}{\delta((1+\delta)-z)} \leq \frac{(2\delta+1)z}{\delta(1+\delta)}.$$

Proof: Fix $\delta > 0$. Define $f : [0,1] \to \mathbf{R}$ by

$$f(z) = \frac{(2\delta+1)z}{\delta(1+\delta)} - \ln \frac{(z+\delta)(1+\delta)}{\delta((1+\delta)-z)}.$$

We have

$$\begin{aligned} f'(z) &= \frac{2\delta+1}{\delta(1+\delta)} - \left(\frac{\delta((1+\delta)-z)}{(z+\delta)(1+\delta)}\right)\left(\frac{\delta((1+\delta)-z)(1+\delta)+(z+\delta)(1+\delta)\delta}{\delta^2(1+\delta-z)^2}\right) \\ &= \frac{2\delta+1}{\delta(1+\delta)} - \frac{2\delta+1}{(z+\delta)(1+\delta-z)} \\ &\geq 0. \end{aligned}$$

Thus, f is monotonically increasing and is thus minimized when $z = 0$. The fact that $f(0) = 0$ then completes the proof. □

2.2 The basic family of learning algorithms

The basic family of learning algorithms $\{A_\delta : \delta \geq 0\}$ is designed to perform well on the set of linear functions defined on $[0,1]^n$ whose coefficients are nonnegative and sum to 1. These functions can be viewed as computing weighted averages. Intuitively, the larger δ is the more robust the algorithm is against noise, and, correspondingly, the more slowly the algorithm learns.

The Algorithm A_δ may be stated formally as follows. We maintain a vector of normalized weights which is updated at the end of each trial. For each t, let $\vec{v}_t \in [0,1]^n$ be the algorithm's weights before trial t. When given the instance $\vec{x}_t = (x_{t,1}, ..., x_{t,n}) \in [0,1]^n$ at trial t, the algorithm predicts with $\lambda_t = \vec{v}_t \cdot \vec{x}_t$. Let $\rho_t \in [0,1]$ be the response at trial t.

We initialize the weight vector to $\vec{v}_{1,i} = 1/n$ for all i. At the end of each trial we update the weights as follows:

If $\delta > 0$, our update is

The following series of lemmas also give approximations for quantities arising in our analysis.

Lemma 3: *For all* $x, y \in \mathbf{R}$,

$$x(x-y) \geq \frac{1}{2}(x^2 - y^2).$$

Proof: Suppose $x \geq y$. Then x is at least the average of x and y, which is $(x+y)/2$. Thus,

$$x(x-y) \geq \frac{1}{2}(x+y)(x-y) = \frac{1}{2}(x^2 - y^2).$$

Now, suppose $x < y$. Then $x - y < 0$, and thus the fact that $(x+y)/2 > x$ in this case implies that

$$x(x-y) > \frac{1}{2}(x+y)(x-y) = \frac{1}{2}(x^2 - y^2),$$

completing the proof. □

Lemma 4: *For all* z, δ *and* x *such that* $\delta > 0$, $0 < z \leq 1$ *and* $0 \leq x \leq 1 - z$,

$$\ln \frac{(x+z+\delta)(1-x+\delta)}{(x+\delta)(1-x-z+\delta)} \leq \ln \frac{(z+\delta)(1+\delta)}{\delta(1-z+\delta)}.$$

Proof: Fix $z, \delta > 0$. Define $f : [0, 1-z] \rightarrow \mathbf{R}$ by

$$f(x) = \ln \frac{(x+z+\delta)(1-x+\delta)}{(x+\delta)(1-x-z+\delta)}.$$

Note that it is sufficient to prove that f is convex over its domain, since the right hand side of the claimed inequality is $f(0) = f(1-z)$.

Define $g : [0, 1-z] \rightarrow \mathbf{R}$ by

$$g(x) = \ln \frac{x+z+\delta}{x+\delta}.$$

Then

$$\begin{aligned} f(x) &= g(x) + g((1-z)-x) \\ f'(x) &= g'(x) - g'((1-z)-x) \\ f''(x) &= g''(x) + g''((1-z)-x). \end{aligned}$$

Hence, the result follows from the convexity of g, which is easily verified. □

were too large. Of course, these changes are reversed when the aggregate prediction is too small.

Our algorithms use the above philosophy of updating the weights with the additional crucial feature that the smaller the aggregate error, the "gentler" the updates. In particular, if the aggregate prediction is correct, the weights are not changed.

As is done in [Littlestone, 1989b] for linear threshold functions, we use the relative entropy between our weights and a target set of weights as a measure of progress. The relative entropy is an information theoretic notion normally used to measure the distance between probability distributions.

2.1 Preliminaries

We will find it convenient to discuss sequences $\vec{x}_1, \vec{x}_2, ...$ of elements of $\mathbf{R}^n$. In such circumstances, we will denote the ith component of $\vec{x}_t$ by $x_{t,i}$.

Suppose $\vec{\mu}, \vec{v} \in [0,1]^n$ are such that $||\vec{\mu}||_1 = ||\vec{v}||_1 = 1$. We define the *entropy* of $\vec{\mu}$ to be $\sum_{i=1}^{n} -\mu_i \ln \mu_i$, where $0 \ln 0$ is taken to be 0, and denote this quantity by $H(\vec{\mu})$. The *relative entropy* between $\vec{v}$ and $\vec{\mu}$, denoted by $I(\vec{\mu}||\vec{v})$, is given by

$$I(\vec{\mu}||\vec{v}) = \sum_{i=1}^{n} \mu_i \ln \frac{\mu_i}{v_i}.$$

For any two such $\vec{\mu}$ and $\vec{v}$, it is well known that $I(\vec{\mu}||\vec{v}) \geq 0$ and that $I(\vec{\mu}||\vec{v}) = 0$ iff $\vec{\mu} = \vec{v}$.

We will need the following simple lemmas. The first is due to Kullback [Kullback, 1967].

Lemma 1 ([Kullback, 1967]): *For* $\lambda, \rho \in [0,1]$ $I((\rho, 1-\rho)||(\lambda, 1-\lambda)) \geq 2(\lambda - \rho)^2$.

We will also make use of the following.

Lemma 2 ([Littlestone, 1989b]): *For all* $\beta > 0, x \in [0,1]$,

$$\beta^x \leq 1 + (\beta - 1)x,$$

with equality iff $x = 0$ *or* $x = 1$.

the advisor would be to initially weigh all opinions equally, and adjust the weight assigned to each economist based on her performance.

When using a weighted average for prediction, a natural interpretation of the weights is as the relative "credibilities" of the economists. Given this interpretation, a natural reweighting strategy is to reduce the weights of each economist according to some monotone function of how far off her estimate was (e.g., the Weighted Majority algorithm [Littlestone and Warmuth, 1989]), and then normalize so that the weights sum to one. In the discrete case this approach can lead to logarithmic total mistake bounds [Littlestone, 1988] [Littlestone, 1989b] [Littlestone and Warmuth, 1989]. Furthermore, it was shown in [Littlestone and Warmuth, 1989] that in the continuous case the loss of the advisor is at most $O(\log n)$ plus a constant times the least individual loss of any of the n economists.[4]

However, if one wishes to learn a linear combinations without assuming that any one economist does well individually, then this strategy does not work. Suppose that there were three economists: one who always wildly overestimated the GNP, one who wildly underestimated the GNP, and one who always gave an estimate slightly greater than the correct GNP. Suppose further that the average of the estimates of the two wild economists was always exactly correct, so that there was a weighting with zero total loss. It is easy to see that in this example the loss of the above strategy is unbounded: the wild economists' contribution will be steadily decreased and in the limit the prediction of the economist who is always slightly off will dominate.

It turns out that the following intuition can be translated into an essentially optimal learning algorithm. If the aggregate opinion was greater than the true GNP, then those whose predictions were too small were "pulling" the aggregate in the right direction, and the marginal effect of increasing their weights is to improve the aggregate prediction, even if their predictions were very inaccurate. Thus one would want to increase the weights of those whose predictions were too small, and decrease the weights of those whose predictions

[4] Again, these results are with respect to the loss function $|\lambda_t - \rho_t|$.

rule including experimental comparisons is given in [Cesa-Bianchi *et al.*, 1991].

Our algorithms are motivated by the algorithms of [Littlestone, 1988] [Littlestone, 1989b] for learning simple boolean functions, such as clauses with a small number of literals. In that case the predictions and responses are boolean. A mistake occurs when the prediction and response disagree, and the loss is taken to be the total number of mistakes in all trials. Algorithms are given in those papers for learning k-literal clauses whose worst case mistake bounds are at most a constant factor from optimal. We generalize the techniques developed there to the learning of linear functions defined on $\mathbf{R}^n$. Algorithms for a simple continuous-valued case which are within a constant factor of optimal have already been given in [Littlestone and Warmuth, 1989]. In our notation, this is the case when exactly one of the hidden μ_i's is 1 and the rest are 0.[3]

As in [Littlestone, 1988] [Littlestone, 1989b] [Littlestone and Warmuth, 1989] and the Widrow-Hoff rule [Widrow and Hoff, 1960] [Duda and Hart, 1973], our algorithms maintain a vector of n weights that is updated each trial after the response is received. Let $\vec{v}_t$ represent this weight vector before trial t. Our algorithms always predict with the current weight vector: i.e., they predict $\lambda_t = \vec{v}_t \cdot \vec{x}_t$. Note that in the noise-free case it is easy to always find a coefficient vector v consistent with the previously observed examples, i.e., such that for all j less than t, $\vec{v} \cdot \vec{x}_j = \rho_j$. However, consistency is neither necessary nor sufficient to obtain the performance we describe. We show that an algorithm that predicts using an arbitrary consistent linear function can have loss of $\Omega(n)$ (Theorem 11). Our algorithms do not necessarily maintain consistency with previously observed examples. Instead, they are designed so that they "learn a lot" from a large loss, so that the cumulative loss is only logarithmic in n instead of linear.

To get some intuition about updates of the weights that might achieve the above, let us go back to our initial example of predicting the GNP. An obvious strategy for the advisor would be to predict with the average estimate of the economists. Suppose, however, the advisor notices that some economists are better at predicting the GNP. A good method for

[3]These results are with respect to the loss function $|\lambda_t - \rho_t|$.

the corresponding weighted average of economists' estimates always equals the actual GNP. For that case, we describe a family $\{A_\delta : \delta > 0\}$ of learning algorithms. We show that for any finite sequence of trials and any $\delta > 0$, the loss of A_δ is bounded by $O(\min\{\ln n - H(\vec{\mu}) + \sum_{t=1}^{m}(\vec{\mu} \cdot \vec{x}_t - \rho_t)^2\})$, where the minimum[1] is over all probability vectors $\vec{\mu} \in [0,1]^n$. In particular, this implies that the total loss of A_δ is $O(\log n + N)$, where N is the total loss obtained from the best fixed weight vector. This performance is obtained even though the algorithm is not given any information about future examples and about the error term (the sum in the above expression). As in the case in which all examples are consistent with some hidden function, we can show that our algorithms are optimal to within a constant factor. We can also give algorithms for more general linear functions defined on more general domains by transforming such problems into the basic problem discussed above. These transformations resemble those studied in [Haussler, 1989b] [Kearns *et al.*, 1987] [Littlestone, 1988] [Pitt and Warmuth, 1990].

It was shown in [Cesa-Bianchi *et al.*, 1991] that the worst-case total loss of the Widrow-Hoff rule (also sometimes called the Δ-rule) [Widrow and Hoff, 1960] [Duda and Hart, 1973] in the setting of this chapter is $\Omega(n + N)$, where, again, N is the total loss of the best fixed weight vector. This contrasts with the bound of $O(\log n + N)$ for our algorithm. On the other hand, techniques due to Mycielski[2] [Mycielski, 1988] can also be used to show that the Widrow-Hoff rule is within a constant factor of optimal for a closely related problem, where, instead of assuming that the hidden weight vector $\vec{\mu}$ consists of nonnegative components summing to one, one assumes that it has Euclidian length at most one, and instead of choosing instances $\vec{x}_1, \vec{x}_2, \ldots$ from $[0,1]^n$, one assumes that the Euclidian length of the instances is 1 [Cesa-Bianchi *et al.*, 1991]. The bound of the sum of squared errors obtained is $2.25(1 + N)$. A more detailed comparison of our algorithm to the Widrow-Hoff

[1] There is a subtle trade off between the two summands in the minimum. Even if there is a $\vec{\mu}$ such that $\rho_t = \vec{\mu} \cdot \vec{x}_t$ for all $1 \leq t \leq m$, the minimum sometimes occurs at a $\vec{\mu}'$ with higher entropy for which $\sum_{t=1}^{m}(\vec{\mu}' \cdot \vec{x}_t - \rho_t)^2 > 0$.

[2] Mycielski gives worst case bounds on the total loss of the Widrow-Hoff rule. Instead of giving bounds in terms of $\sum_{t=1}^{m}(\vec{\mu} \cdot \vec{x}_t - \rho_t)^2$, he states his bounds in terms of $m \max_t(\vec{\mu} \cdot \vec{x}_t - \rho_t)^2$.

2. On-Line Learning of Linear Functions

Suppose, for budget purposes, each year each member of a panel of economists predicts the next year's GNP and an advisor to the president wishes to combine their predictions to obtain a single prediction. If we measure the loss for each year as the square of the difference between the advisor's prediction and actual GNP, a reasonable goal for the advisor is to minimize the worst case total loss over the years. In this chapter, we present near-optimal strategies for combining opinions in situations like this, assuming that some fixed weighted average of the economists is always reasonably close to the actual GNP, which, for problems like this, appears reasonable.

Let CONVEX$_n$ be the class of functions f defined on $\mathbf{R}^n$ by $f(\vec{x}) = \vec{\mu} \cdot \vec{x}$, where $\vec{\mu} \in [0,1]^n$ has components which sum to 1 (let's call such $\vec{\mu}$ "probability vectors" from here on). Note that each function in CONVEX$_n$ takes a different convex combination ("weighted average") of the components of its argument. In this chapter we will concern ourselves with LC$_2$(CONVEX$_n$, N). As we will see later, it is interesting to consider CONVEX$_n$ not only for situations like combining the opinions of experts, where it is interesting for its own sake, but also because the analysis of CONVEX$_n$ forms the basis for the analysis of several natural, and larger, classes of functions.

Let us begin with the case $N = 0$, i.e., the case in which there is an unknown $f \in$ CONVEX$_n$ for which $f(\vec{x}_t)$ always equals ρ_t. We describe an algorithm A_0 for this case, and prove that the worst case sum of squared errors of A_0 on any sequence of trials consistent with an element of CONVEX$_n$ is at most $(\ln n - H(\vec{\mu}))/2$ where $H(\vec{\mu}) = -\sum_{i=1}^{n} \mu_i \ln \mu_i$ is the entropy of the hidden coefficient vector $\vec{\mu}$ that defines f. Since for all relevant $\vec{\mu}$, $H(\vec{\mu}) \geq 0$, another upper bound on total loss of A_0 is $(\ln n)/2$. Also, as $\vec{\mu}$ approaches $(1/n, 1/n, ..., 1/n)$, $H(\vec{\mu})$ approaches $\ln n$, and our bounds approach 0. We show that for all values of $H(\vec{\mu})$, A_0 is optimal to within a constant factor. Note that our bounds hold for an arbitrarily large number m of trials.

Now, suppose that $N > 0$, e.g., that there is not any fixed set of weights such that

Fix X and Y and a learning algorithm A. For a finite sequence of examples $S = \langle (x_t, \rho_t) \rangle_{1 \leq t \leq m}$ we then have that the *prediction* λ_t of A on the t-th *trial* satisfies

$$\lambda_t = A(((x_1, \rho_1), ..., (x_{t-1}, \rho_{t-1})), x_t).$$

For $p \geq 1$, the *p-loss* of A on S is defined as follows:

$$\mathrm{L}_p(A, S) = \sum_{t=2}^{m} |\lambda_t - \rho_t|^p.$$

Note that we begin summing on the second trial. This is reasonable, since an algorithm's prediction on the first trial is made without seeing any examples, and is therefore not an indication of learning ability. The upper bounds of Chapter 2 also hold if we begin summing on the first trial, but not those of Chapter 3. The p-loss of A on a particular trial t is $|\lambda_t - \rho_t|^p$. Finally, if $\mathcal{F}$ is a class of functions from X to Y, let $\mathrm{L}_p(A, \mathcal{F}, N)$ be the supremum of $\mathrm{L}_p(A, S)$ over all finite sequences $S = \langle (x_t, \rho_t) \rangle_{1 \leq t \leq m}$ of examples (of unbounded length) such that there exists $f \in \mathcal{F}$ with $\sum_{t=1}^{m} (f(x_t) - \rho_t)^p \leq N$. $\mathrm{L}_p(A, \mathcal{F}, N)$ measures the algorithm A's ability to take advantage of the fact that a nearly functional relationship from the known class $\mathcal{F}$ exists between the x_t's, which it uses for prediction, and the ρ_t's, which it is trying to predict. The parameter N indicates how close to a function in $\mathcal{F}$ this relationship is. The p-learning complexity of $\mathcal{F}$ (with N noise) is given by

$$\mathrm{LC}_p(\mathcal{F}, N) = \inf_A \mathrm{L}_p(A, \mathcal{F}, N).$$

The p-learning complexity is the best "p-performance" that can possibly be obtained for $\mathcal{F}$ (and N), and therefore gives us a measure of the power of the assumption that there is a function in $\mathcal{F}$ that (nearly) maps x_t's to ρ_t's.

Finally, when $N = 0$, we will drop mention of N from our notation. That is,

$$\begin{aligned} \mathrm{L}_p(A, \mathcal{F}, 0) &= \mathrm{L}_p(A, \mathcal{F}) \\ \mathrm{LC}_p(\mathcal{F}, 0) &= \mathrm{LC}_p(\mathcal{F}). \end{aligned}$$

In Chapter 3, we will consider classes of functions of a single variable designed to capture the intuition that, for many relationships of practical interest, similar inputs tend to yield similar outputs. We show that for several settings of this type, extremely simple algorithms are optimal.

1.1 Defining adversarial learning

Some standard notation and mathematical definitions are listed in Appendix A. We give more topic-specific notation and definitions here.

Let X be a set, $Y \subseteq \mathbf{R}$. We assume that learning takes place in a sequence of *trials*, where in the tth trial,

- The (on-line) learning algorithm receives $x \in X$ from the environment.
- The learning algorithm outputs a prediction $\lambda_t \in Y$ (interpreted as a prediction of the upcoming response ρ_t).
- The learning algorithm receives a response $\rho_t \in Y$.

Note that each pair (x_t, ρ_t) serves a dual role in this setting. At time t, it is used to test the algorithm's predictive ability. For trials $s > t$, it is used by the algorithm to make future predictions.

In keeping with the second role, we define an *example* for (X, Y) to be an element of $X \times Y$. If (x, ρ) is an example, call x the instance and ρ the correct response to x. If f is a function from X to Y, we say that f is *consistent* with an example (x, ρ) if $f(x) = \rho$, and that f is consistent with a sequence S of examples if it is consistent with each example of S.

Each prediction of an on-line learning algorithm (for (X, Y)) is determined by the previous examples and the current instance. Associated with an on-line learning algorithm A we define a mapping of the same name from $(X \times Y)^* \times X$ to Y. Let $\mathcal{A}(X, Y)$ be the set of such mappings corresponding to learning algorithms for (X, Y).

[Maass and Turan, 1989] [Maass and Turan, 1990]. We will find that several techniques developed during the study of mistake-bounded learning, especially some of Littlestone's [Littlestone, 1988,Littlestone, 1989b], are useful for the problems addressed in this part.

Despite the popularity of the mistake-bound model, perhaps the dominant model of learning relationships between $\{0,1\}$-valued quantities is Valiant's PAC model [Valiant, 1984], and variants thereof (esp., [Haussler *et al.*, 1990]). This model, which will be described in more detail in Part II, includes probabilistic assumptions on the learner's environment. Kearns, Li, Pitt, and Valiant [Kearns *et al.*, 1987] and Angluin [Angluin, 1988] independently showed that a "good" learning algorithm in the mistake-bound model can be transformed into a "good" algorithm in the PAC model, and Haussler [Haussler, 1988] showed that in many cases, no transformation was necessary. Littlestone [Littlestone, 1989a] described a transformation which, in some cases, yielded better PAC learning algorithms. Blum [Blum, 1990c] then showed that no such conversion could exist from the PAC model to the mistake-bound model, exhibiting a class which was "learnable" in the PAC model, but not in the mistake bound model.

Littlestone has sinced generalized his transformation to show that algorithms with good performance in the model considered in this part can be transformed to obtain algorithms that are very good in random environments, in a natural sense [Littlestone, 1991]. Thus, "worst-case" analyses like those in this part are interesting not only because they can be applied in a broader variety of circumstances, but also due to their consequences concerning learning in random environments.

In Chapter 2, we will consider learning in this model where it is assumed that the mapping to be learned is linear. An intuitively obvious algorithm is to simply hypothesize at any given time the function which would have yielded the best predictions, had we used it in the past. We show that this algorithm can perform significantly far from optimal in a natural setting, and describe an algorithm whose performance is within a constant factor of optimal. We will see that our results may also be applied when the mapping is a low order polynomial.

1. Introduction

In this part, we will be concerned with situations in which a learner wishes to use the current value of a certain quantity (or quantities) to predict the future value of another quantity. Examples include using the barametric pressure to predict rainfall, using the interest rate to predict changes in the Dow Jones average, or combining the predictions of several experts (e.g., National Basketball Association scouts) on the future value of any quantity (say the scoring average during the first NBA season of a current college senior). We wish to further focus our attention on problems where the value that was predicted is later received, e.g. through observation or measurement, as is the case in the examples sketched above. Finally, we assume that a (nearly) functional relationship exists between the quantity used for prediction and that predicted, and that the learner knows of a class of functions containing the mapping to be "learned."

A main distinguishing feature of the research described in this part is the absence of probabilistic assumptions about the learner's environment. Instead, we assume that the learner's environment is an adversary operating within a certain reasonable constraint. An interpretation of the adversary's constraint is that it enforces that the learner's prior knowledge of the class of functions containing that to be learned is (nearly) accurate. It is a well-established "pseudo-theorem" that nontrivial learning in the absence of such prior knowledge is impossible.

The learning model of this part was introduced by Mycielski [Mycielski, 1988], and independently by Littlestone and Warmuth [Littlestone and Warmuth, 1989].[1] It generalizes the "mistake-bound" model of Angluin [Angluin, 1987] and Littlestone [Littlestone, 1988], in which it is assumed that the quantity to be predicted takes on one of two values. This model and its close relatives have been heavily studied [Angluin, 1988] [Blum, 1990b] [Blum, 1990a] [Blum *et al.*, 1991] [Helmbold *et al.*, 1990] [Littlestone, 1988] [Littlestone, 1989b] [Littlestone and Warmuth, 1989] [Maass, 1991]

[1] These papers will be discussed further in Chapter 2.

Part I

Learning Real-Valued Functions in an Adversarial Environment

students, who have taught me a lot. I'd like to especially thank Naoki Abe, Nicolo Cesa-Bianchi, Yoav Freund, Lisa Hellerstein, Anders Krogh, Nick Littlestone, Aleks Milosavljevic, Giulia Pagallo and Madhukar Thakur.

Finally, I'd like to thank David Haussler and Manfred Warmuth for their generous financial support, which came from ONR grants numbered N00014-86-K-0454 and N00014-91-J-1162. I'd also like to thank UCSC for giving me a Chancellor's dissertation-year fellowship.

Acknowledgements

I'd like to begin by thanking my parents, Ralph and Linda Long, who inspired by their example, provided a loving home, and who dedicated themselves to my and my brother's education. I'd also like to thank my dear friend Melanie Liu, who generously gave emotional support, listened patiently to many descriptions of the work described in this thesis, and asked many interesting questions. Also, thanks to my brother Al Long, with whom I've had many stimulating discussions about this work and related topics, and who has also been very supportive.

Very special thanks go to David Haussler, Dave Helmbold and Manfred Warmuth, who shared a majority of the responsibility for my graduate education. They were very patient, and extremely generous with both time and ideas. Studying with them has been the single greatest joy of my life.

I'd also like to thank those with whom I pursued the work described in this thesis: Shai Ben-David, Nicolo Cesa-Bianchi, David Haussler, David Helmbold, Don Kimber, Nick Littlestone, and Manfred Warmuth. Thank you for the pleasure of working with you, and for allowing me to include our joint research in my thesis. The results of Chapter 2 were obtained through joint work with Nick Littlestone and Manfred Warmuth, and appeared in a preliminary form in [Littlestone *et al.*, 1991]. Chapter 3 contains work done jointly with Don Kimber [Kimber and Long, 1992]. The work described in Chapter 5 was done jointly with Shai Ben-David, Nicolo Cesa-Bianchi, and David Haussler [Ben-David *et al.*, 1992]. David Haussler also contributed to the work contained in Chapter 6 [Haussler and Long, 1990]. Chapter 7 contains joint work with David Helmbold, and improves on the preliminary results we reported in [Helmbold and Long, 1991].

Thanks also to those who contributed indirectly to this thesis by teaching interesting classes. Among the most memorable are George Exner, Sol Friedberg, David Haussler, Dave Helmbold, Phokion Kolaitis, Bill Miller and Manfred Warmuth.

I'd like to thank the Santa Cruz Machine Learning postdocs and my fellow graduate

Towards a More Comprehensive Theory of Learning in Computers

Philip M. Long

ABSTRACT

We attempt to determine the theoretical boundaries of the ability of computers to learn. We consider several rigorous models of learning, aimed at addressing types of learning problems excluded from earlier models.

In Part I, we consider learning dependencies between real-valued quantities in situations where the environment is assumed to be an adversary, operating within constraints that model the prior knowledge of the learner. While our assumptions as to the form of these dependencies is taken from previous work in statistics, this work is distinguished by the fact that the analysis is worst case.

In Part II, we consider learning in situations in which the learner's enviroment is assumed to be at least partially random. We consider methods for extending the tools for learning $\{0,1\}$-valued functions to apply to the learning of many-valued and real-valued functions. We also study the learning of $\{0,1\}$-valued functions in situations in which the relationship to be learned is gradually changing as learning is taking place.

List of Figures

Contents

University of California
Santa Cruz

Towards a More Comprehensive Theory of Learning in Computers

A dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Computer and Information Sciences

by

Philip M. Long

June 1992

The dissertation of Philip M. Long is approved:

David Haussler

David P. Helmbold

Manfred K. Warmuth

Dean of Graduate Studies and Research