Query by Committee

H. S. Seung*
Racah Institute of Physics and
Center for Neural Computation
Hebrew University
Jerusalem 91904, Israel
seung@mars.huji.ac.il

M. Opper†
Institut für Theoretische Physik
Justus-Liebig-Universität Giessen
D-6300 Giessen, Germany
manfred.opper@
physik.uni-giessen.dbp.de

H. Sompolinsky
Racah Institute of Physics and
Center for Neural Computation
Hebrew University
Jerusalem 91904, Israel
haim@galaxy.huji.ac.il

Abstract

We propose an algorithm called *query by committee*, in which a committee of students is trained on the same data set. The next query is chosen according to the *principle of maximal disagreement*. The algorithm is studied for two toy models: the high-low game and perceptron learning of another perceptron. As the number of queries goes to infinity, the committee algorithm yields asymptotically finite information gain. This leads to generalization error that decreases exponentially with the number of examples. This in marked contrast to learning from randomly chosen inputs, for which the information gain approaches zero and the generalization error decreases with a relatively slow inverse power law. We suggest that asymptotically finite information gain may be an important characteristic of good query algorithms.

1 Introduction

Although query algorithms have been proposed for a variety of learning problems[Bau91], little work has gone into understanding the general principles by which these algorithms should be constructed. In this work, we argue that the Shannon information of a query can be a suitable guide[Fed72]. We further show that the degree of disagreement among a *committee* of learners can serve as an estimate of this information value. The algorithms considered in this work focus on minimizing the number of queries required, and hence are most relevant to situations in which queries carry the heaviest computational cost.

We consider the paradigm of *incremental* query learning, in which the training set is built up one example at a time. We restrict our scope to parametric learning models with continuously varying weights, learning perfectly realizable, boolean-valued target rules. The prior distribution on the weight space is assumed to be flat.

An incremental learning procedure consists of two components: a training algorithm and a query algorithm. Given a set of P examples, the *training algorithm* produces a set of weights satisfying the training set. The *query algorithm* is then used to select example $P+1$. Then the training algorithm is run again on the newly incremented training set, and so on.

In this paper, the only training algorithm that we consider is the (zero temperature) Gibbs algorithm, which selects a weight vector at random from the *version space*, the set of all weight vectors that are consistent with the training set. This will enable us to use techniques from statistical mechanics[SST92].

After training $2k$ students on the same training set, the *query by committee* algorithm selects an input that is classified as positive by half of the committee, and negative by the other half. By maximizing disagreement among the committee, the information gain of the query can be made high. In the $k \to \infty$ limit, each query bisects the version space, so that the information gain saturates the bound of 1 bit per query.

In the following, the query by committee algorithm is first illustrated using a very simple model, the high-low game. We then move on to a more complicated model, perceptron learning of another perceptron[GD89]. For both models, the information gain approaches a finite value as the number of queries goes to infinity. This asymptotically finite information gain leads to generalization error that decreases exponentially with the number of queries.

This is in marked contrast to the case of learning with random inputs, in which the information gain

*Present address: AT&T Bell Laboratories, 600 Mountain Ave., Murray Hill, NJ 07974, seung@physics.att.com

†Present address: Institut für Theoretische Physik, Julius Maximilians Universität Würzburg, Am Hubland, D-8700 Würzburg, Germany, opper@vax.rz.uni-wuerzburg.dbp.de

COLT'92-7/92/PA,USA
© 1992 ACM 0-89791-498-8/92/0007/0287...$1.50

approaches zero as the number of examples increases. In the random input case, the generalization error decreases relatively slowly, as an inverse power law in the number of examples.

2 The information content of a query

Denote the target function, or "teacher," by $\sigma_0(\mathbf{X})$, and the parametric learning model, or "student," by $\sigma(\mathbf{W};\mathbf{X})$. Both teacher and student are boolean-valued functions, i.e. maps into the set $\{\pm 1\}$. We further assume that the target function is perfectly realizable by the student, which means that there exists some weight vector $\mathbf{W}_0$ such that $\sigma_0(\mathbf{X}) = \sigma(\mathbf{W}_0;\mathbf{X})$ for all $\mathbf{X}$. The input vector $\mathbf{X}^t$ is called a *positive* or *negative* example depending on the sign of the teacher's output $\sigma^t = \sigma_0(\mathbf{X}^t)$. The *training set* of input-output pairs $\xi^t = (\mathbf{X}^t, \sigma^t)$ determines the *version space*

$$\mathcal{W}_P = \{\mathbf{W} : \sigma(\mathbf{W};\mathbf{X}^t) = \sigma^t, \quad t = 1,\ldots,P\} , \qquad (1)$$

which is the set of all $\mathbf{W}$ consistent with the training set. If the prior distribution $\mathcal{P}_0(\mathbf{W})$ is assumed to be flat, then the posterior distribution is uniform on the version space, and vanishes outside[TLS89]. This is written as

$$\mathcal{P}(\mathbf{W}|\xi^1,\ldots,\xi^P) = \begin{cases} V_P^{-1}, & \mathbf{W} \in \mathcal{W}_P , \\ 0, & \text{otherwise}, \end{cases} \qquad (2)$$

where V_P is the volume of $\mathcal{W}_P$. We consider the Gibbs training algorithm[HKS91], in which the weight vector $\mathbf{W}$ is drawn at random from this posterior distribution.

A *query algorithm* specifies a way of choosing another input $\mathbf{X}^{P+1}$ with which to query the teacher. In general, it can be written as a conditional probability

$$\mathcal{P}(\mathbf{X}^{P+1}|\xi^1,\ldots,\xi^P) , \qquad (3)$$

which we leave unspecified for now. An input $\mathbf{X}^{P+1}$ is chosen from this distribution and receives a label σ^{P+1} from the teacher. The results of this query are then added to the training set.

The entropy of the posterior distribution (2) on the weight space is

$$S = \log V_P . \qquad (4)$$

Since the entropy quantifies our uncertainty about $\mathbf{W}$, the information gained from query $P+1$ can be defined as the reduction in the entropy, or

$$I_{P+1} = -\Delta S = -\log \chi_{P+1} . \qquad (5)$$

Here we have defined the volume ratio

$$\chi_{P+1} \equiv \frac{V_{P+1}}{V_P} . \qquad (6)$$

The information gain I_{P+1} depends on the query sequence $\{\xi^1,\ldots,\xi^{P+1}\}$, although the dependence is not explicit in the notation of (5). This dependence can be eliminated by considering averaged quantities. For example, one can average the information gain (and other quantities of interest) over query sequences, and then over the prior distribution of $\mathbf{W}_0$.

For our purposes, only a partial average will suffice. Holding the query sequence $\xi^1,\ldots,\xi^P$ constant, we will average over the input $\mathbf{X}^{P+1}$ with respect to (3) and the teacher weight vector $\mathbf{W}_0$ with respect to the *posterior* distribution (2). In other words, the average is over all teacher vectors $\mathbf{W}^0$ that are consistent with the first P examples, and over all inputs $\mathbf{X}_{P+1}$ given by the query algorithm. So the average information gain is given by

$$\langle I_{P+1} \rangle = -\langle \log \chi_{P+1} \rangle_{\mathbf{W}_0,\mathbf{X}_{P+1}} \qquad (7)$$

Similarly, we can calculate the complete probability distribution for the volume ratio, which is

$$\mathcal{P}(\chi_{P+1}|\xi^1,\ldots,\xi^P) = \left\langle \delta\left(\chi_{P+1} - \frac{V_{P+1}}{V_P}\right)\right\rangle_{\mathbf{W}_0,X_{P+1}} \qquad (8)$$

Note that these quantities still contain a dependence on the query sequence $\xi^1,\ldots,\xi^P$.

Performing the $\mathbf{W}^0$ average in (7) leads to a bayesian interpretation of the formula. Any input $\mathbf{X}^{P+1}$ divides the version space $\mathcal{W}_P$ into two parts,

$$\mathcal{W}^+ = \{\mathbf{W} \in \mathcal{W}_P : \sigma(\mathbf{W};\mathbf{X}^{P+1}) = +1\} , \qquad (9)$$

$$\mathcal{W}^- = \{\mathbf{W} \in \mathcal{W}_P : \sigma(\mathbf{W};\mathbf{X}^{P+1}) = -1\} . \qquad (10)$$

Averaging over the posterior distribution of $\mathbf{W}_0$, we find that the average information gain (7) is given by

$$\langle I_{P+1} \rangle = \left\langle -\frac{V^+}{V_P}\log\frac{V^+}{V_P} - \frac{V^-}{V_P}\log\frac{V^-}{V_P}\right\rangle_{\mathbf{X}^{P+1}} , \qquad (11)$$

where $V^\pm$ are the volumes of $\mathcal{W}^\pm$ and hence depend upon $\mathbf{X}^{P+1}$ implicitly. After the teacher answers the query, σ^{P+1} is known with certainty. Before the answer arrives, the value of σ^{P+1} is uncertain: according to the bayesian, it is $+1$ with probability V^+/V_P, and -1 with probability V^-/V_P. The entropy of this distribution is precisely the information value of the query, and is the expression inside the average (11). The average information gain is maximized by $\mathbf{X}^{P+1}$ such that $V^+ = V^-$, i.e. by queries that divide the version space in half. In this case of exact bisection, $I = 1$ bit exactly.

Unfortunately, for most nontrivial learning models, the geometry of the version space is complex, and one cannot practically calculate the volumes $V^\pm$ for any given input, much less find an input for which $V^+ = V^-$. Training algorithms typically yield single points in the version space, not global information about the version space. However, a committee of students can be used to obtain global information. Train a committee of $2k$ weight vectors using the Gibbs algorithm. Find an input vector that is classified as a positive example by k members of the committee, and classified as negative by the other k. Query the teacher about this input vector. Train the committee again using the new enlarged training set, and repeat. As $k \to \infty$ this algorithm approaches the bisection algorithm. This algorithm is very much in the spirit of [OH91], in which consensus was used to improve generalization performance. Here we use lack of consensus to choose a query, or a principle of *maximal disagreement*.

We define the generalization function by

$$\epsilon_g(\mathbf{W}, \mathbf{W}_0) = \langle \Theta(-\sigma(\mathbf{W}; \mathbf{X})\sigma(\mathbf{W}_0; \mathbf{X})) \rangle_{\mathbf{X}} , \quad (12)$$

where the average is taken over the prior distribution $\mathcal{P}_0(\mathbf{X})$ of inputs. This measures the probability of error by a student $\mathbf{W}$ on input-output pairs drawn from a teacher $\mathbf{W}_0$. Given a query sequence $\xi^1, \ldots, \xi^P$, the generalization error is defined by

$$\epsilon_g(\xi^1, \ldots, \xi^P) = \langle \epsilon_g(\mathbf{W}, \mathbf{W}_0) \rangle_{\mathbf{W}, \mathbf{W}^0} , \quad (13)$$

where the averages over $\mathbf{W}$ and $\mathbf{W}^0$ are taken over the *posterior* distribution. The average generalization error $\epsilon_g(P)$ is defined as the average of (13) over the query sequence.

3 High-Low

The game of high-low, perhaps the simplest model of query learning, can be formalized within our parametric learning framework. The teacher and student are

$$\sigma_0(X) = \mathrm{sgn}(X - W_0) , \quad (14)$$
$$\sigma(X, W) = \mathrm{sgn}(X - W) . \quad (15)$$

The input and weight spaces are the unit interval $[0, 1]$, and the prior distributions on these spaces are flat,

$$\mathcal{P}_0(W_0) = 1 , \quad (16)$$
$$\mathcal{P}_0(X) = 1 . \quad (17)$$

Given any query X, the teacher will respond with $+1$ or -1, depending on whether X is higher or lower than W.

Suppose that there already exists a training set of $P \geq 2$ examples. Let X_L be the largest negative example, and X_R the smallest positive example. Then the version space is

$$\mathcal{W}_P = [X_L, X_R] , \quad (18)$$

with volume

$$V_P = X_R - X_L . \quad (19)$$

The posterior weight distribution

$$\mathcal{P}_P(W) = \begin{cases} V_P^{-1}, & W \in [X_L, X_R] , \\ 0, & \text{otherwise}, \end{cases} \quad (20)$$

has entropy $S = \log V_P$. The Gibbs training algorithm simply picks a W at random from the version space, i.e. from the interval $[X_L, X_R]$.

We first consider the case of randomly chosen inputs. With probability $1 - V_P$ the next input X^{P+1} will fall outside the version space, so that the volume ratio will be 1. If X^{P+1} falls inside the version space, the probability distribution of the volume ratio is given by 2χ, where this result is obtained by averaging over all $X^{P+1}, W \in [X_L, X_R]$. Combining these two alternatives, we find that the probability distribution of the volume ratio is

$$\mathcal{P}(\chi_{P+1}|V_P) = (1 - V_P)\delta(\chi_{P+1} - 1) + V_P 2\chi_{P+1} . \quad (21)$$

The expected information gain is given by the average of $-\log \chi_{P+1}$ with respect to this distribution, or

$$I_{P+1} = \frac{V_P}{2} \quad \text{(nat)} \quad (22)$$

As the number of examples increases, the volume V_P shrinks, and the information gain tends to zero.

The generalization error (13) is linear in the volume of the version space,

$$\epsilon_g(\xi^1, \ldots, \xi^P) = \frac{1}{3} V_P . \quad (23)$$

Averaging this over all possible training sets, one can show that the average generalization error satisfies

$$\epsilon_g(P) = \frac{2}{3} \frac{1}{P + 2} , \quad (24)$$

an inverse power law in the number of examples.

Maximal information gain can be attained by the bisection algorithm, in which X_{P+1} is chosen to be halfway between X_L and X_R. In this case, the probability distribution of the volume ratio is

$$\mathcal{P}(\chi) = \delta\left(\chi - \frac{1}{2}\right) \quad (25)$$

so that the information gain is 1 bit per query. The volume decreases exponentially,

$$V_P = 2^{-P} . \quad (26)$$

Indeed, our knowledge of the binary representation of W_0 increases by one bit per query.

For a committee with $2k$ members, the probability distribution of the volume ratio is given by

$$\mathcal{P}(\chi) \propto 2\chi \int_0^\chi dy \int_\chi^1 dz \frac{y^{k-1}(1 - z)^{k-1}}{z - y} , \quad (27)$$

where we have omitted the normalization constant. Here we have written $\mathcal{P}(\chi)$ rather than $\mathcal{P}(\chi_P)$, since the distribution is independent of the query history, unlike the random input case (21). Figure 1 shows (27) for various values of k. Note that as k approaches infinity, the curves approach the delta function (25) of the bisection algorithm. The average information gain in nats is given by

$$I(k) = \int_0^1 d\chi \log \chi \mathcal{P}(\chi) \quad (28)$$
$$= \psi(2k + 2) - \psi(k + 2) - \frac{\psi'(k + 2)}{\psi(k + 2) - \psi(k)} + \frac{1}{2} ,$$

where $\psi(x) = \Gamma'(x)/\Gamma(x)$ is the Euler ψ-function. As $k \to \infty$, this saturates the bound of 1 bit per query.

Since $\epsilon_g = V_P/3$ is the product of the volume ratios,

$$\log \epsilon_g = \sum_{t=1}^{P} \log \chi_t + \text{const} . \quad (29)$$

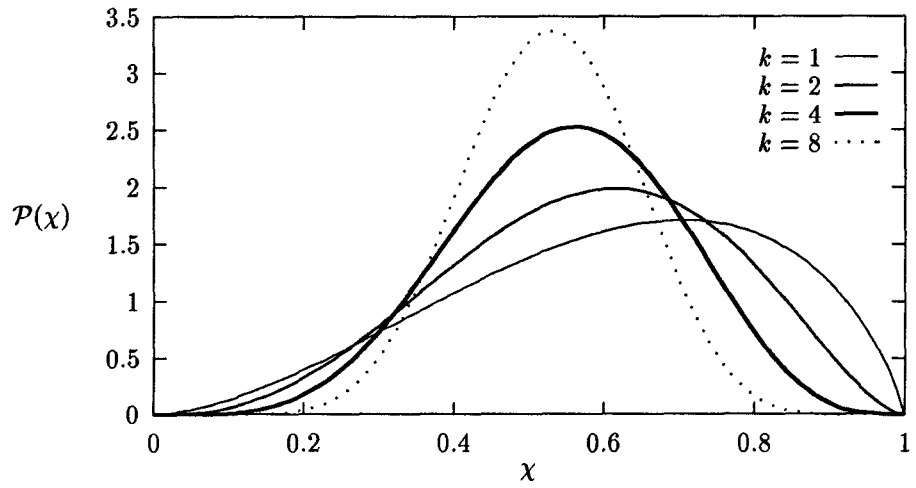

Figure 1: Probability distribution of the volume ratio for a $2k$-member committee playing the high-low game.

The volume ratios χ_t are independent, identically distributed random variables drawn from (27). Hence by the central limit theorem ϵ_g approaches a log-normal random variable as $P \to \infty$. The most probable $\epsilon_g(P)$ scales like

$$\epsilon_g(P) \sim e^{-PI(k)} , \tag{30}$$

where $I(k)$ is given by (28).[1] Thus for query learning of high-low there is a very simple exponential relationship between information gain and generalization error. In the next section we will see that a similar relationship can be written for perceptrons.

4 Perceptron Learning

We now consider perceptron learning of another perceptron, where the teacher and student are given by

$$\sigma_0(\mathbf{X}) = \mathrm{sgn}(\mathbf{W}_0 \cdot \mathbf{X}) , \tag{31}$$

$$\sigma(\mathbf{X}; \mathbf{W}) = \mathrm{sgn}(\mathbf{W} \cdot \mathbf{X}) . \tag{32}$$

The vectors $\mathbf{W}$, $\mathbf{W}_0$, and $\mathbf{X}$ all have N components. The weight space is taken to be the hypersphere

$$\mathcal{W} = \{\mathbf{W} : \mathbf{W} \cdot \mathbf{W} = N\}, \tag{33}$$

and the input distribution is Gaussian

$$\mathcal{P}_0(\mathbf{X}) = (2\pi)^{-N/2} \exp^{-\frac{1}{2}\mathbf{X}\cdot\mathbf{X}} . \tag{34}$$

Given P examples the posterior distribution is uniform on the version space

$$\mathcal{W}_P = \{\mathbf{W} \in \mathcal{W} : (\mathbf{W} \cdot \mathbf{X}^t)(\mathbf{W}_0 \cdot \mathbf{X}^t) > 0, \forall t = 1, \ldots P\} \tag{35}$$

[1] We could also calculate the average of ϵ_g rather than $\log \epsilon_g$. The coefficient of P in the resulting exponential is given by $\log\langle\chi\rangle$ rather than $\langle\log\chi\rangle$.

4.1 Random inputs

When all inputs are chosen at random from the distribution (34), the replica method can be used to calculate the entropy of the posterior distribution[GT90]. The calculation is exact in the thermodynamic limit, where $P, N \to \infty$ with $\alpha = P/N$ constant. The entropy per weight $s \equiv S/N$ is then

$$s(\alpha) = \frac{1}{2}\log(1-q) + \frac{1}{2}q + 2\alpha \int Dx\, H(\gamma x) \log H(\gamma x) , \tag{36}$$

where

$$\gamma \equiv \sqrt{\frac{q}{1-q}} , \tag{37}$$

$$Dx \equiv \frac{dx}{\sqrt{2\pi}} e^{-x^2/2} , \tag{38}$$

$$H(y) \equiv \int_y^\infty Dx . \tag{39}$$

The entropy must be extremized with respect to the order parameter q, from which the average generalization error can be obtained using the relation

$$\epsilon_g = \pi^{-1} \cos^{-1} q . \tag{40}$$

In the large α limit, this leads to the inverse power law behavior

$$\epsilon_g(\alpha) \approx \frac{0.625}{\alpha} \tag{41}$$

The large α asymptotics can also be obtained by examining the scaling of the entropy with the generalization error, similar to the arguments in [SST92] using the microcanonical high-T limit for a general classification of learning curves. As $q \to 1$, the first term of (36)

dominates, so that the entropy has a simple logarithmic dependence on the generalization error

$$s \approx \log \epsilon_g \ , \tag{42}$$

where we have used the asymptotic result $\epsilon_g \approx \pi^{-1}\sqrt{2(1-q)}$. The information gain is given by

$$\begin{aligned} I(\alpha) = -\frac{\partial s}{\partial \alpha} &= -2 \int Dx\, H(\gamma x) \log H(\gamma x) \\ &\approx -2\sqrt{1-q} \int dx H(x) \log H(x) \\ &\approx 1.60 \epsilon_g \ , \end{aligned} \tag{43}$$

This condition combined with (42) can only be satisfied by the inverse power law $\epsilon_g \sim 1/\alpha$, as in (41).

We have performed a Monte Carlo simulation of this algorithm. The deterministic perceptron algorithm[MP88] was used to obtain a weight vector in the version space. Then zero-temperature Monte Carlo was used to random walk inside the version space. To maintain acceptance rates of approximately 50%, the size of the Monte Carlo step was scaled downward with increasing α. The resulting learning curve is shown in Fig. 2, and fits the analytic results obtained through the replica method.

4.2 Query by committee

Because each query depends on the previous history of queries, the committee algorithm is a dynamical process, where the number of examples plays the role of "time." In the replica calculations for this algorithm, we must know the overlaps between weight vectors at different "times." This makes the calculations much more difficult than for the case of random-inputs. The replica calculation in the appendix makes the simplifying assumption that the typical overlap between weight vectors at different "times" is equal to the typical overlap of two vectors at the earlier "time." In other words, we assume that the overlaps satisfy

$$\frac{\mathbf{W}^t \cdot \mathbf{W}^{t'}}{N} = q(\min\{t, t'\}) \ , \tag{44}$$

and are self-averaging quantities. Taking the thermodynamic limit $N \to \infty$ turns the discrete-time dynamics in t into a continuous-time dynamics in $\alpha \equiv t/N$. The entropy of the weight space at time α is calculated by treating $q(\alpha')$ for previous "times" $\alpha' < \alpha$ as an external field. The resulting saddle point equation for $q(\alpha)$ is an integral equation that can be solved numerically.

The replica results for the two-member committee are shown in Fig. 2, along with Monte Carlo simulations. In the simulations, zero-temperature Gibbs training was implemented by training two perceptrons using the standard perceptron algorithm, and then equilibrating them with zero-temperature Monte Carlo. Input vectors were then selected at random from the prior distribution (34) until an input that produced disagreement was found. The teacher was queried on this input, and the input and the teacher's output were added to the training set.

	(nat)	(bit)
2	0.523	0.754
4	0.592	0.854
6	0.621	0.896
∞	0.693	1

Table 1: Information gain of the query by committee algorithm for perceptron learning

Note that the generalization error of the two-member committee is very close to the random input results for α of order 1 or less. Even for relatively large α, different committee sizes (not shown) perform almost identically. However, the algorithms are quite different in their large α asymptotic properties. By taking the derivative of the committee entropy (66) with respect to α, we find that the information gain approaches the limit

$$I(\alpha) \to -2 \frac{\int dx H^k(x) H^k(-x) H(x) \log H(x)}{\int dx H^k(x) H^k(-x)} \tag{45}$$

as $\alpha \to \infty$. These limiting values for some small k can be found in Table 1. As $k \to \infty$, the information gain saturates the bound of one bit per query.

How does the information gain affect the generalization performance? For the committee algorithm, we have found that the information gain approaches a finite constant value

$$\frac{ds}{d\alpha} \to -I(\infty) \tag{46}$$

We assume that the entropy is still given by $s(\alpha) \approx \log \epsilon_g$ as in (42). This assumption, which is shown to be consistent in the appendix, leads to the asymptotic result

$$\epsilon_g \sim e^{-\alpha I(\infty)} \ . \tag{47}$$

The generalization error is asymptotically *exponential* in α, with a decay constant determined by the information gain. This is markedly faster than the inverse power law (41) for random inputs.

In our implementation of the committee algorithms, a source of random inputs is screened by a committee until one is found that provokes disagreement. A practical drawback of this scheme is that the time to find a query diverges like the inverse of the generalization error. In this respect, algorithms which construct queries directly may be superior. For example, the query algorithm proposed by Kinzel and Ruján[KR90, WR92] for perceptron learning constructs input vectors that are perpendicular to the student weight vector. In conjunction with the Gibbs training algorithm, this method yields generalization performance only slightly worse than that of the committee algorithms for moderate α, but much faster query times. However, the committee algorithms appear to possess a generality that other query algorithms do not.

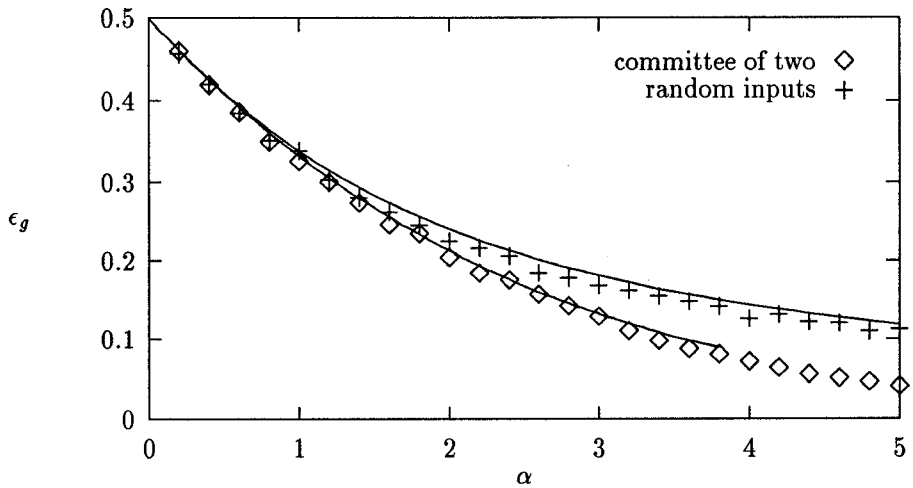

Figure 2: Generalization curves for perceptron learning using the random-input, committee, and minimal training set algorithms. The Monte Carlo simulations were done with $N = 25$, averaged over 64 samples. After each query, the perceptron was equilibrated for 1024 steps. The standard error of measurement is smaller than the size of the symbol. The solid lines are analytic results from replica theory.

5 Conclusion

We have compared query by committee and random-input training for the case of the perceptron. For random inputs, the information gain approaches zero with the generalization error,

$$I(\alpha) \equiv -\frac{ds}{d\alpha} \sim \epsilon_g \ . \tag{48}$$

For the committee algorithm, the information gain is asymptotically finite

$$I(\alpha) \to I(\infty) \ , \tag{49}$$

where $I(\infty) > 0$ is given by (45).

For perceptron learning, the entropy behaves asymptotically as

$$s \equiv \frac{S}{N} \sim \log \epsilon_g \ , \tag{50}$$

for both random-input and query algorithms.

Given $ds/d\alpha$ and s in terms of ϵ_g, one can derive the generalization curve as a function of α. For random inputs, (48) and (50) imply

$$\epsilon_g \sim \frac{1}{\alpha} \ . \tag{51}$$

For the committee algorithms, (49) and (50) imply

$$\log \epsilon_g \sim -I(\infty)\alpha \ . \tag{52}$$

The high-low game and the perceptron have provided simple illustrations of the query by committee algorithm, and of the relationship between information gain and generalization error. However, we have not addressed the issue of whether the behaviors exhibited by these toy models are general. For what architectures does query by committee lead to asymptotically finite information gain? Under what conditions does asymptotically finite information gain lead to exponential generalization curves? These questions are currently under investigation.

Acknowledgements

We acknowledge helpful discussions with Y. Freund, M. Griniasty, D. Hansel, N. Tishby, and T. Watkin.

References

[Bau91] E. Baum. Neural net algorithms that learn in polynomial time from examples and queries. *IEEE Trans. in Neural Networks*, 2:5–19, 1991.

[Fed72] V. V. Fedorov. *Theory of Optimal Experiments.* Academic Press, New York, 1972.

[GD89] E. Gardner and B. Derrida. Three unfinished works on the optimal storage capacity of networks. *J. Phys.*, A22:1983–1994, 1989.

[GT90] G. Györgyi and N. Tishby. Statistical theory of learning a rule. In W. K. Theumann and R. Köberle, editors, *Neural Networks and Spin Glasses*, pages 3–36, Singapore, 1990. World Scientific.

[HKS91] D. Haussler, M. Kearns, and R. Schapire. Bounds on the sample complexity of bayesian

learning using information theory and the VC dimension. In M. K. Warmuth and L. G. Valiant, editors, *Proceedings of the Fourth Annual Workshop on Computational Learning Theory*, pages 61–74, San Mateo, CA, 1991. Morgan Kaufmann.

[KR90] W. Kinzel and P. Ruján. Improving a network generalization ability by selecting examples. *Europhys. Lett.*, 13:473–477, 1990.

[MP88] M. L. Minsky and S. Papert. *Perceptrons.* MIT Press, Cambridge, expanded edition, 1988.

[OH91] M. Opper and D. Haussler. Generalization performance of bayes optimal classification algorithm for learning a perceptron. *Phys. Rev. Lett.*, 66:2677–2680, 1991.

[SST92] H. S. Seung, H. Sompolinsky, and N. Tishby. Statistical mechanics of learning from examples. *Phys. Rev.*, A45:6056–6091, 1992.

[TLS89] N. Tishby, E. Levin, and S. Solla. Consistent inference of probabilities in layered networks: Predictions and generalization. In *Proc. Int. Joint Conf. on Neural Networks*, volume 2, pages 403–409, Washington, DC, 1989. IEEE.

[WR92] T. L. H. Watkin and A. Rau. Selecting examples for perceptrons. *J. Phys.*, A25:113–121, 1992.

Appendix: Replica calculations for query by committee

The volume V_P of the version space determines the entropy $S = \ln V_P$ for a fixed teacher $\mathbf{W}_0$. The average of the entropy over all possible query sequences $\xi^1, \ldots, \xi^P$ consistent with the teacher is denoted by S_{av}. From this quantity we shall find the overlap q between two random weight vectors inside the version space $\mathcal{W}_P$. If we take one them to be the target vector, this in turn will give us the *generalization error* on a new random input.

We begin with the definition

$$S_{av} = \langle \ln V_P \rangle_{\{\mathbf{X}^t\}} \quad (53)$$

$$= \lim_{n\to 0} \frac{\partial}{\partial n} \ln \langle \langle V_P^n \rangle_{\{\mathbf{X}^t\}} \rangle_{\mathbf{W}_0} \quad (54)$$

In the second line we have introduced the replica trick, and an average of $\mathbf{W}_0$ over the prior distribution of teachers has been added, as in [OH91]. This makes the symmetry of the teacher and the students explicit, so that the teacher vector $\mathbf{W}_0$ can be treated as another replica,

$$S_{av} = \lim_{n\to 0} \frac{\partial}{\partial n} \ln \operatorname*{Tr}_{\{\sigma^l\}} \int \prod_{\mu=0}^{n} d\mathbf{W}_a \times \left\langle \prod_{t=1}^{P} \prod_{\mu=0}^{n} \Theta(\sigma^t \mathbf{W}_\mu \cdot \mathbf{X}^t) \right\rangle_{\{\mathbf{X}^t\}} \quad (55)$$

Note that for all integer n we have now $n+1$ replicas.

Recall that for the $k = 2$ committee, the $t+1$st input $\mathbf{X}^{t+1}$ is selected so as to cause disagreement between two committee members $\mathbf{W}_+^t$ and $\mathbf{W}_-^t$ drawn at random from the version space $\mathcal{W}_t$. Thus the probability density for $\mathbf{X}^{t+1}$ is

$$\mathcal{P}(\mathbf{X}^{t+1}|\mathbf{W}_+^t, \mathbf{W}_-^t) \propto \mathcal{P}_0(\mathbf{X}^{t+1}) \times \sum_{\sigma=\pm 1} \Theta(\sigma \mathbf{W}_+^t \cdot \mathbf{X}^{t+1}) \Theta(-\sigma \mathbf{W}_-^t \cdot \mathbf{X}^{t+1}) \quad (56)$$

where a prefactor must be added to achieve the correct normalization $\int d\mathbf{X}^{t+1} \mathcal{P}(\mathbf{X}^{t+1}|\mathbf{W}_+^t, \mathbf{W}_-^t) = 1$. The prior distribution of patterns $\mathcal{P}_0(\mathbf{X})$ is the Gaussian

$$\mathcal{P}_0(\mathbf{X}) = (2\pi)^{N/2} e^{-\frac{1}{2}\mathbf{X}\cdot\mathbf{X}} . \quad (57)$$

Let us consider the average over $\mathbf{X}^{t+1}$, with the committee weights $\mathbf{W}_+^t$ and $\mathbf{W}_-^t$ fixed for the time being.

$$\left\langle \prod_{\mu=0}^{n} \Theta(\sigma^{t+1} \mathbf{W}_\mu \cdot \mathbf{X}^{t+1}) \right\rangle_{\mathbf{X}^{t+1}}$$

$$= \int d\mathbf{X}^{t+1} \prod_{\mu=0}^{n} \Theta(\sigma^{t+1} \mathbf{W}_\mu \cdot \mathbf{X}^{t+1}) \mathcal{P}(\mathbf{X}^{t+1}|\mathbf{W}_+^t, \mathbf{W}_-^t)$$

$$= \int d\mathbf{X}^{t+1} \mathcal{P}_0(\mathbf{X}^{t+1}) \prod_{\mu=0}^{n} \Theta(\sigma^{t+1} \mathbf{W}_\mu \cdot \mathbf{X}^{t+1}) \times \sum_{\sigma=\pm 1} \Theta(\sigma \mathbf{W}_+^t \cdot \mathbf{X}^{t+1}) \Theta(-\sigma \mathbf{W}_-^t \cdot \mathbf{X}^{t+1}) \times \quad (58)$$

$$\left(\sum_{\sigma=\pm 1} \int d\mathbf{X} \Theta(\sigma \mathbf{W}_+^t \cdot \mathbf{X}) \Theta(-\sigma \mathbf{W}_-^t \cdot \mathbf{X}) \mathcal{P}_0(\mathbf{X}) \right)^{-1}$$

The average with respect to $\mathcal{P}_0(\mathbf{X}^{t+1})$ involves $n+3$ quantities $\mathbf{W}_a \cdot \mathbf{X}^t/\sqrt{N}$, where a can be any of the symbols $0, 1, \ldots, n, +, -$. These quantities are Gaussian random variables with zero mean, and covariance given by

$$\left\langle \frac{\mathbf{W}_a \cdot \mathbf{X}}{\sqrt{N}} \frac{\mathbf{W}_b \cdot \mathbf{X}}{\sqrt{N}} \right\rangle_{\mathcal{P}_0(\mathbf{X})} = \frac{\mathbf{W}_a \cdot \mathbf{W}_b}{N} = q_{ab} . \quad (59)$$

The average (58) depends on the vectors $\mathbf{W}_a$ only through these overlaps q_{ab}. In accord with the ansatz (44) and the ansatz of replica symmetry, we assume

$$q_{\mu\nu} = \delta_{\mu\nu} + (1 - \delta_{\mu\nu}) q \quad (60)$$

$$q_{\mu\pm} = q(t) \quad (61)$$

$$q_{\pm\pm} = 1 \quad (62)$$

$$q_{\pm\mp} = q(t) . \quad (63)$$

We now make the replacements

$$\frac{\mathbf{W}_\pm^t \cdot \mathbf{X}^{t+1}}{\sqrt{N}} \to x\sqrt{q(t)} + z_\pm \sqrt{1 - q(t)} , \quad (64)$$

$$\frac{\mathbf{W}_\mu \cdot \mathbf{X}^{t+1}}{\sqrt{N}} \to x\sqrt{q(t)} + y\sqrt{q - q(t)} + z_\mu \sqrt{1 - q} . \quad (65)$$

where $x, y, z_\mu, z_\pm$ are uncorrelated Gaussian random variables of unit variance. This change of variables yields the proper covariances.

This average must be done for every $\mathbf{X}^t$, and yields an entropy that depends on the order parameters $q(t)$, $t = 1$ to $P-1$, and on the overlap q at time P of the weight vectors $\mathbf{W}_\mu$. We treat the $q(t)$ as external fields and determine q variationally. This is valid provided that the $q(t)$ are self-averaging quantities, so that the fluctuations in the committee members at times $t < P$ do not affect the calculation of the entropy at time P. To take the continuum limit $N \to \infty$, we define $P = \alpha N$ and $t = \alpha' N$ and replace the product $\prod_{t=1}^{P}(\ldots)$ by $\exp(N \int_0^\alpha d\alpha' \ln(\ldots))$. Finally, the $\mathbf{W}_\mu$ integrals are done, as usual, by introducing entropic terms containing q.

Generalizing to a committee of $2k$ members, we find for the entropy per weight

$$s = \frac{1}{2}q + \frac{1}{2}\ln(1-q) + 2\int_0^\alpha d\alpha' \frac{\int Dx \int Dy H^k(\gamma' x) H^k(-\gamma' x) H(u) \ln H(u)}{\int Dx H^k(\gamma' x) H^k(-\gamma' x)}, \tag{66}$$

where

$$\gamma' \equiv \sqrt{\frac{q(\alpha')}{1-q(\alpha')}}, \tag{67}$$

$$u \equiv x\sqrt{\frac{q(\alpha')}{1-q}} + y\sqrt{\frac{q-q(\alpha')}{1-q}}. \tag{68}$$

The value of $q = q(\alpha)$ is determined variationally from (66).

The information gain can be calculated by differentiating (66) with respect to α,

$$I(q) = -2\frac{\int Dx H^k(\gamma x) H^k(-\gamma x) H(\gamma x) \log H(\gamma x)}{\int Dx H^k(\gamma x) H^k(-\gamma x)}, \tag{69}$$

where γ is defined as in (37). As $q \to 1$, this yields the asymptotic result (45).

Assuming that the generalization error is asymptotically exponential in α, one can show that the disorder term in (66), the integral over α', approaches a constant. Hence the $\ln(1-q)$ term dominates, $s \sim \log \epsilon_g$, and the derivation in the text of asymptotically exponential ϵ_g is self-consistent.