Proceedings of the Third NEC Research Symposium:
"Computational Learning and Cognition" (SIAM, 1993)

Chapter 10
Theoretical Issues in Learning from Examples

H. Sompolinsky*

Abstract

This paper reviews some of the recent developments in the theory of supervised learning within the framework of statistical mechanics. The main focus of the paper is the properties of *zero temperature* learning which selects at random one of the parameter sets that minimize the training error. The main results concerning the shapes of the learning curves are summarized and discussed. Several outstanding issues are discussed, including the evolution of the architecture of the learning systems during training, and the role of the input distribution. New results concerning learning in multi-layer networks, which illustrate these issues, are reported.

1 Introduction

One of the interesting developments in the theory of supervised learning has been the formulation of a statistical mechanical (SM) framework of learning from examples. This framework has not only defined new models of learning, but also provided powerful analytical tools to study their performance. Using the replica method and mean-field theory, new results regarding the shapes of learning curves have been derived. The SM formulation of the problem and its main general properties are summarized in Section 2. In Section 3, I summarize the current understanding of the shapes of learning curves in various classes of systems. Sections 2 and 3 are based largely on the work described in detail in [34, 32]. For earlier work on the SM of learning from examples see [10, 36, 16, 25, 18, 15, 17]. I have also included in Section 3 several extensions of previous results, as well as a discussion on the relation between generalization error and entropy. Most of the results reported in Section 3 and in the following sections are within the framework of the zero temperature limit of Gibbs learning.

Usually one views learning as a process which determines the values of the parameters of a system with a given architecture, e.g., size and number of layers. However, in some cases, the architecture itself may be effectively modified by the learning process. Section 4 discusses the evolution

*Racah Institute of Physics and Center for Neural Computation, Hebrew University, Jerusalem 91904, Israel.

of the system's architecture during learning. This is done in the context of *convergent* multi-layer networks, where the size of the output layer is much smaller than that of the input layer. In such architectures, the relatively few weights close to the output do not contribute significantly to the total entropy. The modification of these weights amounts to modification of the effective size and architecture of the network. An example of learning in a two layer network is presented. Plausible modification of the naive Gibbs distribution to bias certain architectures is discussed.

Valiant's *probably almost correct* (PAC) learning model [37, 9] is manifestly independent of the form of the distribution of the inputs on which the system operates. In contrast, in the SM framework, learning is defined relative to a specific input distribution which is usually assumed to be "well behaved". Nevertheless, relatively little attention has been given to the influence that the shape of this distribution might have on the easiness of learning a task. Section 5 discusses the effect of the shape of the input distribution on the number of examples needed to learn a classification task. An example of perceptron learning with an input distribution that consists of a *Gaussian mixture* is presented. Sections 4 and 5 contain a preliminary presentation of new results that will be reported in detail elsewhere [4, 5]. Concluding remarks are given in Section 6.

2 Statistical Mechanics of Learning from Examples

2.1 General Framework

We consider here a deterministic system that operates on an M-dimensional input space according to $\sigma = \sigma(\mathbf{W};\mathbf{S})$. Here $\mathbf{S}$ is an M-component vector S_i $(i = 1,\ldots,M)$, representing the inputs to the system, and $\mathbf{W}$ denotes the N parameters W_i $(i = 1,\ldots,N)$ that specify the system. The output of the system is denoted by σ which for simplicity will be assumed to be a single real number. Using the language of neural networks we will call W_i the "weights" of the system. The weight space is characterized by a prior measure $d\mu(\mathbf{W})$ defined on the N-dimensional space of $\mathbf{W}$.

The goal of learning from examples is to find a set of W_i that yields a good approximation to a *target* function $\sigma_0(\mathbf{S})$. The system is provided with a set of *examples* consisting of P input-output pairs $(\mathbf{S}^l, \sigma_0(\mathbf{S}^l))$, with $l = 1,\ldots,P$. We assume that each input vector $\mathbf{S}^l$ is chosen at random from the entire input space according to some normalized *a priori* measure denoted $d\mu(\mathbf{S})$.

Learning in the framework of statistical mechanics is based on the *training energy*

$$E(\mathbf{W}) = \sum_{l=1}^{P} \epsilon(\mathbf{W};\mathbf{S}^l)\ , \tag{2.1}$$

where the *error function* $\epsilon(\mathbf{W};\mathbf{S})$ is zero if $\sigma(\mathbf{W};\mathbf{S}) = \sigma_0(\mathbf{S})$ and positive otherwise. The performance of a given network $\mathbf{W}$ on the whole input space

is measured by the *generalization function*, defined as the average error of the network over the whole input space, i.e.,

$$\epsilon(\mathbf{W}) = \int d\mu(\mathbf{S})\, \epsilon(\mathbf{W};\mathbf{S}) \,. \tag{2.2}$$

We distinguish between learning of *realizable rules* and *unrealizable rules*. Realizable rules are those target functions $\sigma_0(\mathbf{S})$ that can be completely realized by at least one of the networks in the weight space. Thus in a realizable rule there exists a weight vector $\mathbf{W}^*$ such that

$$\epsilon(\mathbf{W}^*, \mathbf{S}) = 0, \quad \text{for all } \mathbf{S} \,, \tag{2.3}$$

or equivalently, $\epsilon(\mathbf{W}^*) = 0$. An *unrealizable* rule is a target function for which

$$\epsilon_{\min} = \min_{\mathbf{W}} \epsilon(\mathbf{W}) > 0 \,. \tag{2.4}$$

In the SM framework the learning process generates a Gibbs distribution in the weight space,

$$\mathcal{P}(\mathbf{W}) = Z^{-1} e^{-\beta E(\mathbf{W})} \,, \tag{2.5}$$

where $\beta = T^{-1}$ characterizes the width of the Gibbs distribution. The normalization factor Z is the partition function

$$Z = \int d\mu(\mathbf{W}) \exp(-\beta E(\mathbf{W})) \,. \tag{2.6}$$

Note that $\mathcal{P}(\mathbf{W})$ has been defined relative to the prior measure $d\mu(\mathbf{W})$. This is termed *Gibbs learning*. Its performance in a given task is evaluated by averaging the performance of a given network $\mathbf{W}$ with respect to $\mathcal{P}(\mathbf{W})$. This average is called a thermal average, and will be denoted by $\langle \cdots \rangle_T$. In most cases, the thermally averaged quantities are further averaged with respect to the sampling of examples. This second average, called a quenched average, is denoted by $\langle\!\langle \cdots \rangle\!\rangle \equiv \int \prod_l d\mu(\mathbf{S}^l)$. In particular, the *average training* and *generalization errors* are given by

$$\epsilon_t(T,P) \equiv P^{-1} \langle\!\langle \langle E(\mathbf{W}) \rangle_T \rangle\!\rangle \,, \tag{2.7}$$

$$\epsilon_g(T,P) \equiv \langle\!\langle \langle \epsilon(\mathbf{W}) \rangle_T \rangle\!\rangle \,. \tag{2.8}$$

The entropy S is defined by

$$S(T,P) = -\int d\mu(\mathbf{W}) \langle\!\langle \mathcal{P}(\mathbf{W}) \ln \mathcal{P}(\mathbf{W}) \rangle\!\rangle. \tag{2.9}$$

The entropy per weight will be denoted by $s = S/N$.

2.2 The Thermodynamic Limit

Classical statistical theory of learning from examples is usually concerned with the performance of a system in the limit of a large size of training set, namely, $P \to \infty$, while other parameters, in particular the size of the system, N, are held fixed. Thus, many of the results are applicable only in the limit where the generalization error is near zero or near its minimal value $\epsilon_{\min}$. In contrast, statistical mechanics is useful primarily in the limit of many degrees of freedom, i.e., $N \to \infty$. A main reason for this is the fact that in this limit the distribution of quantities such as ϵ_g and ϵ_t are sharply peaked about their thermal average so that those averages represent the typical behavior. It should be stressed, however, that the SM formulation is well defined for arbitrary system size, and in some cases, useful theoretical results can be derived from it which do not depend on N being large (an example will be mentioned later).

In the case of learning from examples, definition of the thermodynamic limit also requires specification of how the number of examples depend on N. The appropriate limit is $N, P \to \infty$ while the ratio

$$\alpha \equiv P/N \tag{2.10}$$

is held fixed. Note that the temperature T is also held fixed. This definition is reasonable since it is expected that for large N appreciable reduction in ϵ_g requires number of examples that is of the same order as N. ¿From the SM perspective, the above scaling ensures that the energy which is proportional to P, competes with the entropy, which is proportional to N.

Several qualifications to the above scaling assertion are worth mentioning. First, for some problems the above scaling does not hold at zero T. A simple example is a perceptron with N inputs and a *linear output* that learns a rule derived by a similar perceptron with binary weights. If the weights of the *student* are also constrained to binary values then $\epsilon_g(\alpha) = 0$ at $T = 0$ for $\alpha > 0$. In this case, the appropriate scaling of P is probably $P \propto N/(\ln N)^x$ with a positive exponent x (For details see the section on linear perceptron with discrete weights in [32]). Nevertheless, in problems involving dichotomies (i.e., thresholded outputs rather than linear outputs) or in cases where the weights are continuous variables, the above scaling is generically expected to hold for $T = 0$ as well.

Another point is that in general there may be more than one natural way of scaling up the size of the system. Estimating N in real life learning systems is another open issue. In particular, redundancy in the inputs may generate strong correlations in the weights even if *a priori* they are independent variables. Also, the *effective* size of the system that participates in the learning may be smaller than the total size N. This last issue will be discussed in Section 4.

One reason why Gibbs learning is a good model for learning lies in the general property that *for any fixed* T, increasing α leads to a good

generalization, i.e.,

$$\epsilon_g \to \epsilon_{\min} \ , \ \epsilon_t \to \epsilon_{\min} \ , \ \alpha \to \infty \ ; \ T \ fixed \ . \tag{2.11}$$

Another general property of Gibbs learning is concerned with the relation between generalization and training. Using the definitions Eqs. (2.7) and (2.8) we have shown that

$$\epsilon_t(T,\alpha) \le \epsilon_g(T,\alpha) \tag{2.12}$$

for all T and α.

An interesting question is whether reducing T is a good strategy for learning. Thermodynamics ensures that ϵ_t will decrease with T. However, as far as we know, ϵ_g can in principle have a minimum at non-zero T (for fixed α). Cases where ϵ_g does not change with T below some critical value are known. Furthermore, approximate solutions of several models of learning unrealizable rules yield a minimum of $\epsilon_g(T,\alpha)$ with respect to T, at non-zero values of T. Unfortunately, those solutions suffer from an instability, known as replica symmetry breaking instability (see the following paragraph), and it is not known whether the exact solutions will have this property as well. (For details see [32].)

2.3 Theoretical Methods

Here I briefly comment on the two main tools that are used in the theoretical analysis of the properties of the Gibbs distribution, Eq. (2.5). They are the replica method and the mean-field theory. The replica method deals with the evaluation of quenched averages [12]. In particular, the important quantity $\langle\!\langle \ln Z \rangle\!\rangle$ is evaluated using

$$\langle\!\langle \ln Z \rangle\!\rangle = \lim_{n\to 0} \frac{d\langle\!\langle Z^n \rangle\!\rangle}{dn} \ . \tag{2.13}$$

The *replica trick* is to evaluate $\langle\!\langle Z^n \rangle\!\rangle$ for positive integer n , by replicating the system $\mathbf{W}$ (with the same set of examples) n times, and then analytically continue to $n = 0$. For Z of Eq. (2.6) we obtain

$$\langle\!\langle Z^n \rangle\!\rangle = \int \prod_{\sigma=1}^{n} d\mu(\mathbf{W}^\sigma) \exp(-PG[\mathbf{W}^\sigma]) \ , \tag{2.14}$$

where G is

$$G[\mathbf{W}^\sigma] = -\ln \int d\mu(\mathbf{S}) \exp\left(-\beta \sum_{\sigma=1}^{n} \epsilon(\mathbf{W}^\sigma;\mathbf{S})\right) . \tag{2.15}$$

Thus, in the replica formalism the number of examples appears as a prefactor in the exponent of Eq. (2.14). All other example dependence has been removed, so that the replicated energy G depends only on the form

of $\epsilon(\mathbf{W};\mathbf{S})$ and on the nature of the *a priori* measure on the input space, $d\mu(\mathbf{S})$.

To evaluate the thermodynamic limit, one should calculate

$$\lim_{N\to\infty}\lim_{n\to 0} N^{-1} d\langle\!\langle\, Z^n \,\rangle\!\rangle / dn\,.$$

However, in most cases the only feasible procedure is to take first the limit $N \to \infty$ and then $n \to 0$. The interchange of order of limits has been justified for certain spin-glass problems by van Hemmen and Palmer [38, 39]. Most probably, their arguments can be extended to the present class of problems as well.

In many of the learning problems, one can evaluate Eq. (2.14) in the limit $N \to \infty$ by expressing it as an integral over few integration variables (the number of which remains finite in the $N \to \infty$ limit) and evaluating this integral by a saddle point method which, in principle, is exact in the $N \to \infty$ limit. The resultant solution is also known as mean-field theory. The difficulty with this method lies in the extension of $\langle\!\langle\, Z^n \,\rangle\!\rangle$ from positive integer n to real n, and in particular to real n near 0. For any positive integer $n \geq 2$ the correct saddle point is one which does not break the underlying symmetry under permutation of the replica indices. The naive procedure consists of calculating the replica symmetric saddle point for integer $n \geq 2$ and taking the $n \to 0$ limit by treating n as a general real number. However, in many important cases, the replica symmetric theory can be shown to be wrong. In these cases one must make the extension to real $n \leq 2$ using saddle points that break the symmetry of permutation among the replica indices. The saddle point with broken replica symmetry that, in the $n \to 0$ limit, presumably yields the *exact* result for $\lim_{N\to\infty}\langle\!\langle\, N^{-1} \ln Z \,\rangle\!\rangle$, has been studied in detail in the spin-glass problem. The physical implications of the replica symmetry breaking have also been elucidated [27]. These results apply also to problems of learning from examples. However, evaluating the saddle point equations with broken replica symmetry are, except for special cases, extremely difficult. Moreover, adequate justification for the procedures involved is still lacking. Several concrete examples of problems in perceptron learning that involve replica symmetry breaking are described in [34, 32].

2.4 High Temperature Limit

This limit is defined so that both T and α approach infinity, but their ratio remains constant:

$$\beta\alpha = \text{finite}\,, \quad \alpha \to \infty\,, \quad T \to \infty\,. \tag{2.16}$$

In this limit E can simply be replaced by its average $P\epsilon(\mathbf{W})$ as the fluctuations, δE, coming from the finite sample of randomly chosen examples can be ignored. The Gibbs distribution reduces to the following simple form

$$\mathcal{P}(\mathbf{W}) = Z^{-1} \exp(-N\beta\alpha\epsilon(\mathbf{W}))\,, \tag{2.17}$$

$$Z = \int d\mu(\mathbf{W}) \exp(-N\beta\alpha\epsilon(\mathbf{W})) \tag{2.18}$$

where $\epsilon(\mathbf{W})$ is the error function defined in Eq. (2.2). ¿From theoretical point of view, the simplicity of Eq. (2.17) lies in that one does not have to resort to complicated techniques such as the replica method, since quenched averaging is not needed. Note that even in this limit good generalization can be achieved. In particular, as the effective temperature T/α decreases, the network approaches the optimal weight vector $\mathbf{W}^*$, which minimizes $\epsilon(\mathbf{W})$.

Further insight into the nature of the learning in this limit is gained by writing

$$Z = \int d\epsilon \exp N\left(s(\epsilon) - \beta\alpha\epsilon\right) \quad . \tag{2.19}$$

The function $s(\epsilon)$ is the entropy per weight of all the networks with $\epsilon(\mathbf{W}) = \epsilon$, i.e.,

$$s(\epsilon) = N^{-1} \ln \int d\mu(\mathbf{W})\, \delta(\epsilon(\mathbf{W}) - \epsilon) \; . \tag{2.20}$$

In the large N limit the expected generalization error is simply given by

$$\beta\alpha = \partial s/\partial\epsilon \; . \tag{2.21}$$

Thus, the properties of the system in the high-T limit are determined completely by the entropy as a function of the generalization error. Lastly, an important feature of learning at high temperature is the lack of difference between the expected training and generalization errors, i.e., $\epsilon_g(\beta\alpha) = \epsilon_t(\beta\alpha)$.

2.5 Zero T Learning

In the limit of $\beta \to \infty$, the Gibbs distribution collapses to

$$\mathcal{P}(\mathbf{W}) = Z^{-1}\delta(\mathbf{W} - \mathbf{W}_{min}) \tag{2.22}$$

where $\mathbf{W}_{min}$ minimizes the training energy. When the ground state of E is not unique then the zero T learning selects one of these vectors at random.

The zero T limit is useful particularly in realizable problems. Here the ground state of E is zero for all α and the performance depends purely on the statistical properties of the *version space*, namely, the space of all weights that are consistent with the given examples. In the case of a dichotomy (i.e., ± 1 output) the volume of the version space is given by

$$Z = \int d\mu(\mathbf{W}) \prod_{l=1}^{P} \Theta\left(\sigma_0(\mathbf{S}^l)\sigma(\mathbf{W};\mathbf{S}^l)\right) \tag{2.23}$$

where l runs over the P examples. Increasing P improves the performance by shrinking the version space until only the target weight vector remains.

However, this process not only reduces the entropy of the solution space but may also modify its shape. Therefore there is no simple general relationship between S, and ϵ_g even in zero-T learning, as will be discussed below.

In the thermodynamic limit, Gibbs learning in general, and at $T = 0$ in particular, may exhibit rather unexpected behavior. The version space may break up into several effectively disjoint subspaces. In such a case its volume, Eq. (2.23), will be of the form

$$Z = \sum_{k}^{K} Z_k = \sum_{k}^{K} e^{N s_k}. \tag{2.24}$$

In cases where the subspaces are related by a symmetry operation all the entropies s_k may be equal. For instance, fully connected multi-layer networks may possess permutational symmetry, where permuting among the hidden units of each hidden layer generates an equally good set of weights, see [3, 13, 28, 29] and Section 3 below. In the absence of symmetry, generically one s_k will be larger than the rest. The subspace with the maximal entropy will dominate the sum in Eq. (2.24). In other words, with probability approaching unity as $N \to \infty$, the zero T learning algorithm will select W that resides in the subspace with the maximal entropy. The other subspaces represent *metastable* states. Their contribution may still be important in that a learning dynamics which only approximates the zero T learning may end in one of the metastable states. Examples of metastability in perceptron learning and their relevance to the dynamics are discussed in [32]. Realizations of Eq. (2.24) that are specific to multi-layer networks will be discussed in Section 4.

Most of the discussion that follows will be within the framework of zero T learning.

3 Learning Curves: Summary of Results

3.1 Smooth Networks

An interesting class of networks for which a universal behavior of the learning curves for large α exists is the class of smooth networks. They are defined as having continuously varying weights and an error function $\epsilon(\mathrm{W};\mathrm{S})$ that is twice differentiable with respect to W in the vicinity of the optimal weight vector W^*, which minimizes $\epsilon(\mathrm{W})$. At large P the Gibbs distribution is sharply peaked around W^*. Expanding $\ln \mathcal{P}(\mathrm{W})$, we have shown that its form near the optimal vector is Gaussian with a width that scales as $1/\sqrt{P}$. Since the deviation of ϵ_t and ϵ_g from $\epsilon_{\min}$ is quadratic in $\delta\mathrm{W}$ it follows that their tail for large P has a $1/P$ form. In the $T = 0$ limit we have obtained

$$\epsilon_g(P) = \epsilon_{\min} + \left(\frac{\mathrm{Tr} V U^{-1}}{2P}\right) + O(P^{-2}) \,. \tag{3.1}$$

$$\epsilon_t(\alpha) = \epsilon_{\min} - \left(\frac{\mathrm{Tr}VU^{-1}}{2P}\right) + \mathcal{O}(P^{-2}) \,. \tag{3.2}$$

The matrix U_{ij} is the Hessian of the error function at the optimal weight vector $\mathbf{W}^*$, i.e.,

$$U_{ij} = \int d\mu(\mathbf{S})\, \partial_i \partial_j \epsilon(\mathbf{W}^*, \mathbf{S}) \,. \tag{3.3}$$

The symbol ∂_i denotes $\partial/\partial W_i$. The matrix V_{ij} is

$$V_{ij} = \int d\mu(\mathbf{S})\, \partial_i \epsilon(\mathbf{W}^*, \mathbf{S}) \partial_j \epsilon(\mathbf{W}^*, \mathbf{S}) \,. \tag{3.4}$$

The above results predict an important relationship between the expected training and generalization errors at $T = 0$. According to Eqs. (3.1) and (3.2) both errors approach the same limit ϵ with a $1/P$ power law. The coefficients of $1/P$ in the two errors are identical in magnitude but different in sign yielding

$$\frac{\partial \epsilon_t}{\partial P} = -\frac{\partial \epsilon_g}{\partial P} \,, \quad P \to \infty, \quad T = 0 \,. \tag{3.5}$$

This result can be used to estimate the expected generalization error from the measured training error in smooth networks.

Lastly, we would like to emphasize that contrary to some apparent misconceptions [2] our results for smooth networks have been derived by the replica theory using only the limit of large P (and not large N). The derivation is therefore valid for arbitrary system sizes. In the limit of large N we expect that $\mathrm{Tr}VU^{-1}$ scales as N. Therefore, in the thermodynamic limit, Eqs. (3.1) and (3.2) can be viewed also as expansions in $1/\alpha$. Also, in the derivation of Eqs. (3.1) and (3.2) no assumption has been made regarding the nature of the learning system, apart from the smoothness requirements. Thus, our result for smooth networks is to our knowledge the first universal prediction on the asymptotic shapes of learning curves. It demonstrates the usefulness of the replica method.

3.2 Stochastic Machines

An interesting class of learning systems consists of learning a stochastic input-output relation described by the conditional probability $P(\sigma|\mathbf{S};\mathbf{W}_0)$ where $\mathbf{W}_0$ denotes the vector of parameters specifying the target distribution that generates the data. The asymptotic shapes of the learning curves of this system have been derived recently by Amari and Murata [1]. As I show below, this system is an interesting special case of the general results for smooth networks Eqs. (3.1) and (3.2).

Given a set of P input-output pairs one can train the system with the log-likelihood training energy,

$$E(\mathbf{W}) \;=\; -\sum_{l=1}^{P} \ln P(\sigma_l|\mathbf{S}_l;\mathbf{W}) \;. \tag{3.6}$$

This energy is of the form of Eq. (2.1) above with the identification

$$\epsilon(\mathbf{W};\mathbf{S},\sigma) = -\ln P(\sigma|\mathbf{S};\mathbf{W})\,. \tag{3.7}$$

Thus our general results apply here as well. Note that the quenched average consists here of

$$\langle\!\langle \cdots \rangle\!\rangle \equiv \int \prod_l d\mu(\mathbf{S}^l)\, d\sigma^l\; P(\sigma^l|\mathbf{S}^l;\mathbf{W}_0)\ldots . \tag{3.8}$$

In particular if $\ln P$ is twice differentiable near $\mathbf{W}_0$ then these systems fall under the category of smooth networks. Evaluating Eqs. (3.3) and (3.4) and using the normalization property of the conditional probability it is easy to show that

$$U_{ij} = V_{ij} = \langle\!\langle\, (\partial_i \ln P)(\partial_j \ln P)\,\rangle\!\rangle \tag{3.9}$$

where the derivatives are evaluated at the target weights $\mathbf{W}_0$. This matrix is known in statistics as the *Fisher information matrix* [11, 1]. Since U and V are identical, $\mathrm{Tr}VU^{-1} = N$. Hence Eqs. (3.1) and (3.2) reduce to the simple form

$$\epsilon_g(P) = \epsilon_{\min} + \frac{1}{2\alpha} \tag{3.10}$$

$$\epsilon_t(P) = \epsilon_{\min} - \frac{1}{2\alpha} \tag{3.11}$$

in agreement with Amari and Murata [1]. It is interesting that in the case of a log-likelihood error function, the asymptotic behavior of ϵ_g and ϵ_t is independent of the details of the problem. It should be stressed, however, that this simple form holds only when the target distribution is contained within the space of the trained one, i.e., that the architecture of the learning machine contains $\mathbf{W}_0$.

Learning probabilistic rules has been studied also within the framework of PAC learning [21] and the Bayes approach [19].

3.3 Learning Realizable Dichotomies

When the output of the system is thresholded the energy function is not smooth and the above results are not valid. Considerable attention has been given to the rate at which the entropy of the version space decreases as examples are added. The *information gain* per example (in bits) [36, 20] is defined as

$$I(\alpha) \equiv -(\ln 2)^{-1}\partial s/\partial\alpha\ . \tag{3.12}$$

Interesting bounds on ϵ_g for zero T learning have been derived in terms of $I(\alpha)$ [20]. They read

$$h^{-1}(I(P)) \le \epsilon_g(P) \le \frac{1}{2}I(P) \tag{3.13}$$

where h^{-1} is the inverse of the function $h(x) = -(\ln 2)^{-1}(x\ln x + (1-x)\ln(1-x))$, with $0 \le x \le \frac{1}{2}$. Consequences of these bounds will be given below.

- **Dichotomies with continuous weights:** This class of systems consists of learning dichotomies that can be expressed as

$$\sigma(\mathbf{S};\mathbf{W}) = sign\,(f(\mathbf{S};\mathbf{W})) \tag{3.14}$$

where f is a smooth function of $\mathbf{W}$. An example is a multi-layer network with continuously varying weights and sigmoidal hidden units but a thresholded output. This class has been studied recently by Amari et al. [2]. At large α the weights of the system $\mathbf{W}$ will be close to the target vector $\mathbf{W}_0$ and Eq. (3.14) reduces to

$$\sigma(\mathbf{S};\mathbf{W}) \approx sign\,(f(\mathbf{S};\mathbf{W}_0) + \nabla_{\mathbf{W}} f(\mathbf{S};\mathbf{W}_0)\cdot\delta\mathbf{W})) \tag{3.15}$$

where $\delta\mathbf{W} \equiv \mathbf{W} - \mathbf{W}_0$. Thus the errors are restricted to a volume of inputs of a width of the order $|\delta\mathbf{W}|$ around the decision hypersurface $f(\mathbf{S};\mathbf{W}_0) = 0$. Assuming a smooth measure on the input space, one obtains 'roughly'

$$Z \approx \int d(\delta\mathbf{W}) e^{-P|\delta\mathbf{W}|} \approx P^{-N} \tag{3.16}$$

from which it follows that $s \approx -\ln\alpha$ as $P \to \infty$ yielding

$$I(\alpha) \approx \frac{1}{(\ln 2)\alpha} \quad , \quad P \to \infty \,. \tag{3.17}$$

Under these conditions one expects

$$\epsilon_g \approx \frac{N\epsilon_1}{P} \quad , \quad P \to \infty \,. \tag{3.18}$$

The general results, Eq. (3.17) and (3.18), have been derived by Amari et al. [2]. They are consistent with the known specific SM solutions of learning dichotomies in networks with continuous weights [32]. For more details see [2, 33].

At present, the general dependence of the coefficient ϵ_1 on the properties of the system is not known. In fact, substituting Eq. (3.17) into Eq. (3.13) yields

$$\frac{\ln 2}{2|\ln\alpha|} \le \epsilon_1 \le \frac{1}{2(\ln 2)} \,. \tag{3.19}$$

Note that only the upper bound provides a useful bound on ϵ_1.

Equations (3.13) and (3.17) imply that as P increases the information provided by an additional example vanishes. This is a consequence of the *random* sampling of each new example. In contrast, certain algorithms for drawing examples by *queries* [31, 14] have the property that

$$\lim_{P\to\infty} I(P) = I_\infty > 0 \,. \tag{3.20}$$

In fact, this property may serve as a qualitative distinction between random sampling of examples and good query algorithms. For more details see [31, 14].

It is also of interest to consider $I(\alpha)$ and $\epsilon_g(\alpha)$ in the limit of *small* α. Let us assume for simplicity that the target rule divides the input space into two subspaces of equal probability, and likewise that the prior measure on W is unbiased to $\sigma = +1$ or -1. It can then be shown that, for continuous W, the entropy takes the form

$$s(\alpha) \approx s_0 - (\ln 2)\alpha + s_2\alpha^2 \quad , \quad \alpha \to 0 \tag{3.21}$$

where s_0 is the entropy of the prior measure on W and $s_1 > 0$ is a constant that depends on the task and system. Under these conditions one obtains

$$\frac{1}{2} - \epsilon_g \approx \epsilon_2\alpha \quad , \quad \alpha \to 0 \tag{3.22}$$

with $\epsilon_2 > 0$. Using the bounds Eq. (3.13) with

$$I(\alpha) \approx 1 - \frac{s_2\alpha}{\ln 2} \tag{3.23}$$

yields

$$\frac{1}{\sqrt{\alpha}} \leq \epsilon_2 \leq \frac{s_2}{2\ln 2} \, . \tag{3.24}$$

Here again only the upper bound provides a useful bound on ϵ_2. Further study is needed in order to understand the dependence of the coefficients ϵ_1, and ϵ_2 on the system properties, and to derive tight bounds on them.

The relation between entropy and generalization in zero T learning will be discussed again in Section 4.

* **Dichotomies with discrete weights:** The most dramatic departure from the gradual power law predicted above occurs in learning realizable dichotomies where both the target and the space of W consist of only discrete valued weights. An example is a perceptron learning a perceptron dichotomy, $sign(\mathbf{S}\cdot\mathbf{W}_0)$, where all the weights of both 'student' and 'teacher' perceptrons are binary valued. In this case it has been shown that a discontinuous transition to a perfect learning from a state with large ϵ_g to a state with $\epsilon_g = 0$ occurs at a critical value of α. This transition is sharp only in the large N limit. Otherwise it is smeared over a narrow range of α near α_c. The discontinuous transition to perfect learning in single layer [15, 34, 32, 7] and two-layer [35, 28, 22] networks with binary weights has been studied. In other cases involving non-smooth weight space, learning curves may show exponential tails or power laws with powers smaller than -1.

3.4 Learning Unrealizable Dichotomies

Very little is known about the general properties of learning curves in unrealizable non-smooth tasks. All known SM solutions of models of learning unrealizable dichotomies with continuous weights predict a zero T behavior of the form

$$\epsilon_g(\alpha) - \epsilon_{\min} \propto \alpha^{-\frac{1}{2}} . \tag{3.25}$$

Examples are learning by a perceptron a dichotomy that is not linearly separable [18], or a perceptron learning a rule corrupted by noise [17]. Unfortunately, the underlying theories are definitely not exact because they are based on a replica symmetric saddle point, which is unstable in these unrealizable cases. Therefore, it is unclear to what extent the above prediction holds, even for these specific cases. Deriving a general form for the asymptotic shapes of learning curves in learning unrealizable dichotomies with smooth weights remains an interesting open challenge. At present, it is unclear whether PAC theory can yield interesting bounds in the case of unrealizable rules.

4 The Evolution of Architecture in Multi-Layer Networks

Although the learning algorithm superficially treats all the weights equally, different weights may have very different influence on the emergent network, particularly in large multi-layer convergent networks. An example is a fully connected two-layer network with M inputs, K_0 hidden units, and a single output. For simplicity we assume that K_0 is held fixed while M approaches ∞. The first layer weight vectors, i.e., those connecting the inputs to the hidden units are denoted by $\mathbf{J}^k$, $k = 1, ..., K_0$. The second-layer weights are denoted by W^k. In such an architecture the direct entropic contribution of the relatively few W's is negligible. On the other hand their value determines in general the function of the first layer and therefore also the entropy of the whole network. Hence, the averaging over all weights in the version space is effectively broken into

$$Z = Tr_W \exp\left(Ms(W)\right) \tag{4.1}$$

where $Ms(W)$ is the entropy of the first-layer weights for a fixed set of W. Equation (4.1) can be viewed as a sum over all subspaces of the version space corresponding to different effective architectures. ¿From the discussion of Eq. (2.24) it follows that in the limit of large M only the architecture with the maximal entropy will survive.

To illustrate this point, let us consider learning a target rule generated by a two-layer network, also called a *committee machine*:

$$\sigma_0(\mathbf{S}) \;=\; sign\left(\sum_{k=1}^{K} sign(\mathbf{J}_0^k \cdot \mathbf{S})\right) \tag{4.2}$$

where the K vectors $\mathbf{J}_0^k$ are M-dimensional vectors, where M denotes the dimensionality of the input space. Let us assume that the prior weight space in which learning occurs corresponds to all two-layer networks of the form

$$\sigma(\mathbf{S}) = sign\left(\sum_{k=1}^{K_0} W^k \, sign(\mathbf{J}^k \cdot \mathbf{S})\right) \tag{4.3}$$

with $K_0 > K$, $\mathbf{J}^k \cdot \mathbf{J}^k = 1$, and each of the W_k is restricted to the values 0 and 1. Then the weight space consists of all *committee machines* of the form

$$sign\left(\sum_{k=1}^{K'} sign(\mathbf{J}^k \cdot \mathbf{S})\right) \tag{4.4}$$

where $1 \leq K' \leq K_0$. To avoid ambiguities I will assume that all K and K' are odd. Note that those first-layer vectors that are connected to the hidden units with zero W's, i.e., $\mathbf{J}^k$ with $k = K'+1, ..., K_0$, do not affect the output of the network hence their values are not affected by the examples. Committee with $K' = 3$ learning a perceptron rule $K = 1$ has been recently studied in [28]. The case $K = K'$ has been recently studied in [29].

Here I want to address two questions: First, which committee size K' will the zero T learning algorithm choose? Secondly, which size is optimal for generalization? Obviously, the answer may depend on $\alpha \equiv P/M$. For large α, it is clear that K' will be at least equal to K as a smaller committee will not be able to satisfy the given examples. What is the situation for small α?

To investigate these issues, we have calculated the generalization errors and entropies of committee machines of size K' learning at zero T target committee of size K with orthogonal perceptrons, $\mathbf{J}_0^l \cdot \mathbf{J}_0^k = \delta_{kl}$, in the limit of small α [4]. For ϵ_g we have found

$$\epsilon_g(K', K, \alpha) \approx \frac{1}{2} - 2(f_K f_{K'})^2 \alpha \tag{4.5}$$

where

$$f_K \equiv \frac{\sqrt{K}}{\pi 2^{K-1}} \begin{pmatrix} K-1 \\ (K-1)/2 \end{pmatrix} . \tag{4.6}$$

Since f_K is decreasing with K, Eq. (4.5) implies that at small α the optimal architecture from the point of view of generalization is always

$$K' = 1 \tag{4.7}$$

which is a single layer perceptron! This conclusion is independent of the size K of the machine that generated the rule. The moral of this result is the following: when the system is far below the capacity, squeezing the information from the available examples into the smallest network maximizes

the generalization capability. The results for the entropies are less intuitive. We have found at small α

$$S(K';K;\alpha)/M \approx S_0(K_0) - (\ln 2)\alpha + S(K,K')\,\alpha^2 \tag{4.8}$$

where $S_0(K_0)$ is the prior entropy of the space of K_0 vectors $\mathbf{J}^k$, and

$$S(K,K') = f_K^2 - (f_K - f_{K'})^2 \,. \tag{4.9}$$

Note that in Eq. (4.8) we have calculated the total entropy of K_0 vectors $\mathbf{J}^l$, including the prior entropy of the $K_0 - K'$ vectors that are 'passive', as they do not contribute to the output of the network.

Equation (4.9) implies that at small α the committee machine with the maximum entropy is the one that matches that of the target committee. Thus the zero T learning will pick the architecture with

$$K' = K \,. \tag{4.10}$$

The reason for this is as follows: On the one hand, using a small value of K' leaves all the vectors $\mathbf{J}^l$ with $K_0 > l > K'$ out of the learning process, and therefore free from any constraints. On the other hand, the constraint due to the examples that are imposed on the 'active' vectors $\mathbf{J}^l$ with $1 \le l \le K'$ are more stringent than if the learning would have been distributed over a larger network. Balancing the two opposite trends leads to the result Eq. (4.10).

It is interesting to compare these results with the general results for small α, Eqs. (3.21) - (3.24). In particular, the upper bound of Eq. (3.24) would suggest that the optimal ϵ_g will be that with the highest entropy. This is not born out by our results, which say that ϵ_g is optimal for a perceptron, even though in general it is not the architecture with the highest entropy. The reason for the different trends of the entropy and generalization error is the fact that ϵ_g is related to the capacity of the part of the network that participates in the learning. It is therefore unaffected by the presence of large parts which are 'passive'. In contrast, the learning algorithm is in principle affected by the total entropy of the weight space.

The above results suggest a useful modification of the Gibbs learning algorithm. Instead of weighing different architectures with the same *a priori* probability as in Eq. (4.1) one should add priors biasing in favor of small architectures. This can take the form

$$Z = Tr_W \exp\left(M s(W) + M K(W) u\right) \tag{4.11}$$

where $K(W)$ is the number of *active* hidden units or equivalently the number of non-zero second-layer weights. Note that the bias in favor of a small architecture has to be exponential in M to overcome the difference in entropies between the architectures. This modification is related but not equivalent to the common practice of adding *weight decay* terms to the