Optimal learning in multilayer neural networks

O. Winther,[1,*] B. Lautrup,[1,2] and J.-B. Zhang[3]
[1]*CONNECT, The Niels Bohr Institute, Blegdamsvej 17, 2100 Copenhagen, Denmark*
[2]*CERN, Theory Division, 1211 Genève 23, Switzerland*
[3]*Zhejiang Institute of Modern Physics, Zhejiang University, Hangzhou 310027, China*
(Received 7 September 1995; revised manuscript received 3 September 1996)

The generalization performance of two learning algorithms, Bayes algorithm and the ''optimal learning'' algorithm, on two classification tasks is studied theoretically. In the first example the task is defined by a restricted two-layer network, a committee machine, and in the second the task is defined by the so-called prototype problem. The architecture of the learning machine, in both cases, is defined to be a committee machine. For both tasks the optimal learning algorithm, which is optimal when the solution is restricted to a specific architecture, performs worse than the overall optimal Bayes algorithm. However, both algorithms perform better than the conventional stochastic Gibbs algorithm, especially for the prototype problem in which the task and the learning machine are very different.
[S1063-651X(97)04401-2]

PACS number(s): 87.10.+e, 02.50.−r, 64.60.Cn, 84.35.+i

I. INTRODUCTION

Feedforward neural networks are interesting because of their ability to extract an underlying rule from examples [1]. Using the techniques introduced by Gardner and Derrida [2], statistical mechanics has been applied to study how rule extraction takes place in feedforward neural networks (for a review see [3]). In these generalization problems the rule is usually represented by a teacher network, which provides as output the labels of the correctly classified inputs. In recent years interest has moved from the simplest and best understood model, the simple perceptron, to multilayer networks. The simple perceptron is able to implement a limited class of functions, the linearly separable ones, whereas multilayer networks, in principle, are able to approximate any function [4]. Multilayer networks are thus of much greater practical interest.

The statistical-mechanics analysis becomes increasingly involved when an additional hidden layer of processing units is introduced. Our analysis of two-layer networks is limited to the committee machine in which only the weights in the input-to-hidden layer are adjustable and the hidden-to-output connections are fixed to unity. When the outputs of the hidden units are restricted to only ± 1, the output from the committee machine becomes the majority vote of the hidden unit outputs.

In the usual statistical-mechanics approach, training is considered to be a stochastic minimization of an energy function, which for classification problems is taken to be the sum of misclassifications on the training set, a strategy that is usually called *Gibbs learning*. It is possible, however, to get better average generalization ability if we have at our disposal additional knowledge about the rule. It is possible to define an optimal (in the information theoretical sense) learning algorithm, which is the one that gives the lowest average generalization error. This algorithm, the *Bayes algorithm* [5], is defined without reference to the learning machine. It is also possible to define an *optimal learning* algorithm for a specific machine [6]. In this article we analyze optimal learning in the committee machine and the Bayes algorithm for two rules that have been studied in the literature for the case of the Gibbs learning algorithm. From this study it is therefore possible to determine to what degree the interesting multilayer effects observed in the Gibbs learning scenarios are a result of the student network's intrinsic properties (its capacity) or a property of the teacher.

Schwarze [7] and O'Kane and Winther [8] studied learning of two different rules in the fully connected committee machine (the first implemented by another committee machine and the second defined by the so-called proximity task). In both cases it was observed that there exist two learning regimes. For small training sets the solution is symmetric in the sense that all hidden units have equal probability of predicting the right output for the task. In this regime the committee machine cannot do better than the simple perceptron learning the same task. For large training sets a transition to a specialized solution takes place, i.e., the hidden units make a division of labor for the task. Another effect of the symmetry of the rule, called retarded generalization, has been observed by the authors of [9,10] by which up to a certain critical number of examples the learning machine fails to generalize at all.

In Sec. II we outline the statistical approach to learning for the general case of a deterministic binary classifier. In Sec. III we consider the Bayes algorithm and optimal learning in the committee machine for the realizable case of a rule—the teacher—itself defined to be a committee machine of the same structure as the student network. In Sec. IV we consider the Bayes algorithm and optimal learning in the proximity problem, which is only realizable by a committee machine in the limit where the number of hidden units goes to infinity. Finally, in Sec. V we give a summary and a discussion of the results.

II. STATISTICAL THEORY OF LEARNING WITH A TEACHER

The basic information available in the learning problem is the training set: a set of P input-output pairs

*Electronic address: winther@connect.nbi.dk

$D=(S,\tau)=\{\mathbf{S}^\mu,\tau^\mu\}_{\mu=1,\ldots,P}$, where the input is an N-dimensional real vector, $\mathbf{S}=(S_1,\ldots,S_N)$, and the output is binary $\tau=\pm 1$. We shall assume that the examples are obtained from an unknown deterministic classifier (called the teacher), characterized by an output $\tau(V,\mathbf{S})$ and a set of parameters V. The teacher is assumed to draw the examples randomly and independently with this distribution $\mathcal{P}(\mathbf{S}|V)$. As indicated, the distribution may in principle depend on the teacher's internal structure. In Secs. III and IV we present two explicit examples of teachers. The probability that the teacher generates the whole training set becomes

$$\mathcal{P}(D|V)=\prod_{\mu=1}^{P}\Theta[\tau^\mu\tau(V,\mathbf{S}^\mu)]\mathcal{P}(\mathbf{S}^\mu|V). \tag{1}$$

Using Bayesian inversion, we may now calculate the posterior probability of teachers given the training set $\mathcal{P}(V|D)=\mathcal{P}(D|V)\mathcal{P}(V)/\mathcal{P}(D)$, where $\mathcal{P}(V)$ is the *a priori* measure in the space of teachers and $P(D)=\int dV\,\mathcal{P}(D|V)\mathcal{P}(V)$. The actual set of teachers (in V space) that may generate D is called *version space* and is defined as those V's for which $\mathcal{P}(V|D)>0$.

A. Bayes algorithm

We may now calculate the probability in version space for an output σ, given the input $\mathbf{S}$,

$$\mathcal{P}(\sigma|D,\mathbf{S})=\langle\Theta[\sigma\tau(V,\mathbf{S})]\rangle_{(V|D)}. \tag{2}$$

Since we do not know the true rule, except that it must be in the version space somewhere, we may also interpret $\mathcal{P}(\sigma|D,\mathbf{S})$ as the probability that the true rule gives output σ on input $\mathbf{S}$.

Under these circumstances the best we can do is to choose the output label that has the highest probability according to Eq. (2). For binary classification, we have

$$\sigma_{\text{Bayes}}(D,\mathbf{S})=\arg\max_{\sigma}\mathcal{P}(\sigma|D,S)=\operatorname{sgn}\langle\tau(V,\mathbf{S})\rangle_{(V|D)}. \tag{3}$$

This is the *Bayes algorithm*. The probability that the algorithm yields an error is given by $\mathcal{P}(-\sigma_{\text{Bayes}}(D,\mathbf{S})|D,\mathbf{S})$. Averaging over all possible inputs, we obtain the expected generalization error of Bayes algorithm

$$\epsilon_{\text{Bayes}}(D)=\langle\mathcal{P}(-\sigma_{\text{Bayes}}(D,\mathbf{S})|D,\mathbf{S})\rangle_{(\mathbf{S}|V)}. \tag{4}$$

In the subsequent sections we will study the limit of large system size (the thermodynamic limit) in which quantities such as the error are expected to be self-averaging, i.e., $\epsilon_{\text{Bayes}}(D)\approx\epsilon_{\text{Bayes}}\equiv\langle\epsilon_{\text{Bayes}}(D)\rangle_D$. In principle, it is clear how to implement the Bayes classifier using Eq. (3). However, in practical situations it might not be possible to evaluate, i.e., to construct, a learning machine that will implement it. The Bayes algorithm may, nevertheless, serve as a benchmark for all other algorithms.

B. Gibbs learning

In a deterministic binary classifier, such as a feedforward neural network, the input is an N-dimensional real vector $\mathbf{S}=(S_1,\ldots,S_N)$ and the output is given by a binary function $\sigma(W,\mathbf{S})=\pm 1$, where W is a set of structural parameters, called weights, which specify the possible realizations of the general architecture given by the form of the function.

In Secs. III and IV we shall consider a specific two-layer neural network model, the fully connected committee machine student, with N inputs, K hidden units, and a single output unit. The weight vector of the kth hidden unit is denoted by $\mathbf{W}_k$ and the committee machine performs a simple (binary) majority vote on the output from the K simple perceptrons of the hidden layer:

$$\sigma(W,\mathbf{S})=\operatorname{sgn}\left[\frac{1}{\sqrt{K}}\sum_{k=1}^{K}\operatorname{sgn}\left(\frac{1}{\sqrt{N}}\mathbf{W}_k\cdot\mathbf{S}\right)\right].$$

The length of each of the weight vectors is fixed by a spherical constraint $|\mathbf{W}_k|^2=N$ and the components of the inputs are taken to be random, of $O(1)$ and independent. The prefactors $1/\sqrt{N}$ and $1/\sqrt{K}$ are introduced for convenience to make the arguments of the sign functions of $O(1)$.

The network's ability to generalize is measured by the *generalization function*, which is defined to be the error on a single example averaged over all possible input values,

$$\epsilon(W,V)=\langle\Theta(-\sigma(W,\mathbf{S})\tau(V,\mathbf{S}))\rangle_{(\mathbf{S}|V)}, \tag{5}$$

where $\langle\,\rangle_{(\mathbf{S}|V)}=\int d\mathbf{S}\,\mathcal{P}(\mathbf{S}|V)\cdots$ denotes the average over input values. In the Gibbs learning approach the student classifier undergoes training based on minimization of the training error $E(W,D)=\Sigma_{\mu=1}^{P}\Theta(-\tau^\mu\sigma(W,\mathbf{S}^\mu))$. It is assumed that after training the ensemble of student networks will be characterized by a Gibbs posterior probability distribution

$$\mathcal{P}(W|D)=\frac{1}{Z(D)}e^{-\beta E(W,D)}\mathcal{P}(W), \tag{6}$$

where $\mathcal{P}(W)$ is the *a priori* measure in weight space and $T=1/\beta$ is a formal temperature. The normalization constant becomes $Z(D)=\int dW\,\mathcal{P}(W)e^{-\beta E(W,D)}$. The generalization error of the Gibbs algorithm is calculated by taking the posterior average over the generalization function Eq. (5)

$$\epsilon_{\text{Gibbs}}(V,D)=\langle\epsilon(V,W)\rangle_{(W|D)}. \tag{7}$$

This quantity is also expected to be self-averaging in the thermodynamic limit $\epsilon_{\text{Gibbs}}(V,D)\approx\epsilon_{\text{Gibbs}}\equiv\langle\epsilon_{\text{Gibbs}}(V,D)\rangle_{V,D}$.

C. Optimal learning

In the *optimal learning* algorithm we exploit the fact that we can average out the ignorance about the rule in the generalization function to form a new quantity, the *network error* [6],

$$\epsilon_{\text{net}}(W,D)=\langle\epsilon(W,V)\rangle_{(V|D)}. \tag{8}$$

The network error depends only on observable quantities and is thus in principle calculable from the training set and the prior knowledge of the teacher. It is the expected generalization error for any student W that has been presented with the set of examples D.

The best student is, in this case, the one with the lowest network error $W_{\mathrm{opt}}(D)=\arg\min_W \epsilon_{\mathrm{net}}(W,D)$ leading to the smallest expected generalization error

$$\epsilon_{\mathrm{opt}}(D)=\epsilon_{\mathrm{net}}(W_{\mathrm{opt}}(D),D). \tag{9}$$

In contrast to the Bayes algorithm, the optimal learning algorithm does depend on the choice of learning machine. Notice that this is the best that can be done with a fixed student architecture.

In general, one has the following relation between the three learning algorithms described: $\epsilon_{\mathrm{Bayes}}\leqslant\epsilon_{\mathrm{opt}}\leqslant\epsilon_{\mathrm{Gibbs}}$. For the simple perceptron learning a perceptron teacher, it turns out that the equality between the Bayes and optimal learning holds [6]. For the scenarios studied in this paper it turns out not to be so.

III. COMMITTEE MACHINE TEACHER

In the following we consider the learnable case of a task defined by a teacher network of the same structure as the student network $\tau(V,\mathbf{S})=\mathrm{sgn}[(1/\sqrt{K})\Sigma_k \mathrm{sgn}(\mathbf{V}_k\cdot\mathbf{S}/\sqrt{N})]$ trained on $P=\alpha NK$ training examples with inputs drawn component by component with independent normal distributions: $\mathcal{P}(\mathbf{S})=(2\pi)^{-N/2}e^{-(1/2)\mathbf{S}^2}$. The teacher vectors are chosen to be random with spherical normalization, i.e., $\mathcal{P}(V)=\Pi_k\mathcal{P}(\mathbf{V}_k)$ and $\mathcal{P}(\mathbf{V}_k)\propto\delta(|\mathbf{V}_k|^2-N)$.

A. Bayes algorithm

Thus the prior knowledge about the rule that will be used for the Bayes algorithm (and the optimal learning algorithm) consists of the teacher being a committee machine with random weight vectors. In order to calculate the Bayes error it is convenient to rewrite Eq. (4) as

$$\epsilon_{\mathrm{Bayes}}=\langle\Theta(1-2\langle\Theta(\tau(V,\mathbf{S})\tau(V',\mathbf{S}))\rangle_{(V'|D)})\rangle_{(D|V),V,\mathbf{S}},$$

where we have used that $\mathbf{S}$ is independent of V in this context. The Θ function may be expanded as a binomial sum

$$\Theta(1-2P)=\lim_{h\to\infty}\sum_{k=0}^{[h/2]}\binom{h}{k}P^k(1-P)^{h-k},$$

for $P\in\{0,1\}$. In each term of the expansion we are led to evaluate an expression of the form

$$\begin{aligned}y(r)&=\left\langle\prod_{a=1}^{r}\langle\Theta(\tau(V^a,\mathbf{S})\tau(V^0,\mathbf{S}))\rangle_{(V^a|D)}\right\rangle_{(D|V^0),V^0,S}\\&=\int\prod_{a=0}^{r}dV^a\left\langle\prod_{a=0}^{r}\mathcal{P}(V^a|D)\right\rangle_D\\&\quad\times\left\langle\prod_{a=1}^{r}\Theta(\tau(V^0,\mathbf{S})\tau(V^a,\mathbf{S}))\right\rangle_{\mathbf{S}}.\end{aligned}$$

The above expression may be calculated by means of the replica method by introducing $n-r-1$ replicas to take care of $Z(D)^{-r-1}$, where $Z(D)=\int dV\,\mathcal{P}(V)\Pi_\mu\Theta(\tau^\mu\sigma(V,\mathbf{S}^\mu))$ is the zero-temperature partition function [omitting the trivial input prior factor in Eq. (1)]. With the $r+1$ integrals in the numerator it adds up to n replicas and we may therefore conclude that the expression

$$\left\langle\prod_{a=1}^{r}\Theta(\tau(V^0,\mathbf{S})\tau(V^a,\mathbf{S}))\right\rangle_{\mathbf{S}}$$

in the thermodynamic limit will be equal to $y(r)$ with order parameters of the expression taking the saddle-point value of the free energy of the supervised learning problem $-\beta F=\langle\ln Z\rangle_{(V|D),D}$ at zero temperature, which was solved by Schwarze [7] under some symmetry *Ansätze*, which will be described in the following.

Due to the rotational symmetry in the input space, this scalar quantity may depend only on the scalar order parameters $q_{kl}^{ab}=(1/N)\mathbf{V}_k^a\cdot\mathbf{V}_l^b$, where $a,b=0,\ldots,r$ are the replica indices. In the thermodynamic limit these order parameters are self-averaging and the hidden fields $h_k^a=(1/\sqrt{N})\mathbf{V}_k^a\cdot\mathbf{S}$ will be correlated Gaussians with zero mean and $\langle h_k^a h_l^b\rangle_{\mathbf{S}}=q_{kl}^{ab}$. Replica symmetry, i.e., the order parameters independent of the replica indices, is expected to hold for learnable scenarios.

Since K randomly drawn N-component vectors will be mutually orthogonal in the thermodynamic limit with $N\gg K$ it follows directly from our prior choice for the teacher that $q_{kl}^{aa}=\delta_{kl}$. With this prior choice we have not favored any specific correlations between hidden units, so it is natural to assume partial committee symmetry $q_{kl}^{ab}=D+q\,\delta_{kl}$ for $a\neq b$. Using this symmetry *Ansätz* we arrive, for $K\to\infty$, at the result

$$y(r)=2\int Dx\,H^{r+1}\left(\sqrt{\frac{Q_{\mathrm{eff}}}{1-Q_{\mathrm{eff}}}}x\right),$$

where we have defined $Dx=(dx/\sqrt{2\pi})e^{-(1/2)x^2}$, $H(t)=\int_t^\infty Dx$, $Q_{\mathrm{eff}}=2/\pi(d+\arcsin q)$, and $d=KD$, which is assumed to be $O(1)$. We may now evaluate the expansion and reintroduce the Θ function to obtain the simple result

$$\begin{aligned}\epsilon_{\mathrm{Bayes}}&=2\int Dx\,H\left(\sqrt{\frac{Q_{\mathrm{eff}}}{1-Q_{\mathrm{eff}}}}x\right)\\&\quad\times\Theta(1-2H(\sqrt{Q_{\mathrm{eff}}/(1-Q_{\mathrm{eff}})}x))\\&=\frac{1}{\pi}\arccos(\sqrt{Q_{\mathrm{eff}}}).\end{aligned}\tag{10}$$

Before discussing the saddle point of the free energy we will discuss the Gibbs and optimal algorithms for this problem.

B. Gibbs learning

To study the generalization properties of the Gibbs algorithm (and the optimal algorithm) we must calculate the generalization function (5). This was initially done by Schwarze [7] and we will only summarize the results. It is a scalar and may, due to the rotational symmetry in the input space, depend only on the scalar products

$$R_{kl}=\frac{1}{N}\mathbf{W}_k\cdot\mathbf{V}_l,\quad C_{kl}=\frac{1}{N}\mathbf{W}_k\cdot\mathbf{W}_l,$$

where the third possible parameter $(1/N)\mathbf{V}_k\cdot\mathbf{V}_l=\delta_{kl}$ by definition of the problem as discussed above. The assumption of partial committee symmetry used in [7] is

$$R_{kl}=R+\Delta\,\delta_{kl},\quad C_{kl}=C+(1-C)\,\delta_{kl}. \tag{11}$$

This *Ansatz* describes two types of solutions, which are also observed in simulations: the hidden unit permutation symmetric solution with $\Delta=0$ and the specialized solution with $\Delta\neq 0$ in which the symmetry is broken and each hidden unit is correlated with one of the hidden units of the teacher. In the following these solutions will be called the symmetric and the specialized solution, respectively.

The generalization function may be evaluated using the symmetry *Ansatz* (11) and taking $N\gg K\gg 1$,

$$\epsilon(\Delta,\rho,c)=\frac{1}{\pi}\arccos\left(\frac{R_{\mathrm{eff}}}{\sqrt{1+2c/\pi}}\right), \tag{12}$$

where $R_{\mathrm{eff}}=2/\pi(\rho+\arcsin\Delta)$ and the order parameters have been rescaled to $c=KC$ and $\rho=KR$, which are assumed to be $O(1)$. For the Gibbs algorithm in the thermodynamic limit the self-averaging properties means that $R_{kl}=\langle R_{kl}\rangle_{(V|D),(W|D)}$ and $C_{kl}=\langle C_{kl}\rangle_{(W|D)}$ at the saddle point of the free energy. For $T=0$ the average over W corresponds to an average over V and we therefore have $R_{kl}=D+q\,\delta_{kl}$ and $C_{kl}=\delta_{kl}$. The generalization error of the $T=0$ Gibbs algorithm therefore becomes

$$\epsilon_{\mathrm{Gibbs}}=\frac{1}{\pi}\arccos(Q_{\mathrm{eff}}). \tag{13}$$

C. Optimal learning

We shall now show that, in the thermodynamic limit, where the linear extent of version space shrinks as $1/\sqrt{N}$, we may write the average over the generalization function (12) as

$$\epsilon_{\mathrm{net}}(W,D)\equiv\langle\epsilon(R_{kl},C_{kl})\rangle_{(V|D)}$$
$$=\langle\epsilon(\langle R_{kl}\rangle_{(V|D)},C_{kl})\rangle_{(V|D)}+O\left(\frac{K}{\sqrt{N}}\right).$$

The reason lies in the fact that the difference $\Delta\mathbf{V}_k=\mathbf{V}_k-\langle\mathbf{V}_k\rangle_{(V|D)}$ is projected only on $K\ll N$ directions $\mathbf{W}_k$ in R_{kl}. Each coordinate component will be of $O(1)$ on average because $\langle\Delta\mathbf{V}_k^2\rangle_{(V|D)}=N(1-D-q)$; therefore its projection on a particular vector will be $\mathbf{W}_k\cdot\Delta\mathbf{V}_L\approx O(\sqrt{N})$ on the average. Consequently, $\Delta R\approx O(1/\sqrt{N})$.

In order to calculate the optimal student we must minimize the network error (8) with respect to $\mathbf{W}_k$ for all k. Using Lagrange multipliers λ_k to handle the normalization conditions $|\mathbf{W}_k|^2=N$, we find that the optimal student must satisfy

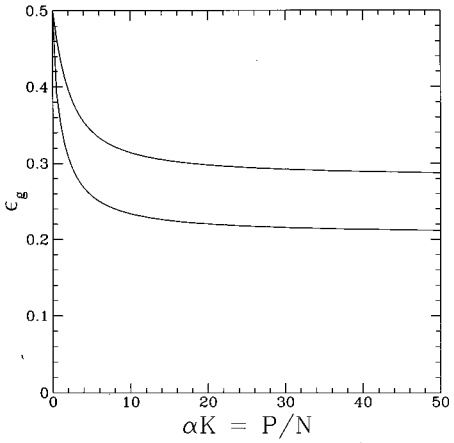

FIG. 1. Learning curve for the committee task on the small-α regime. The lower curve is the result for Bayes algorithm and optimal learning. The upper curve is the zero-temperature Gibbs learning curve.

$$\lambda_k\mathbf{W}_k=\sum_l\left\langle\frac{\partial\epsilon_{\mathrm{net}}(W,D)}{\partial\langle R_{kl}\rangle_{(V|D)}}\right\rangle_{(V|D)}\langle\mathbf{V}_l\rangle_{(V|D)}$$
$$+\sum_{l\neq k}\left\langle\frac{\partial\epsilon_{\mathrm{net}}(W,D)}{\partial C_{kl}}\right\rangle_{(V|D)}\mathbf{W}_l.$$

This shows that the optimal student weights $\mathbf{W}_k$ are linear combinations of the teacher weight vector averages $\langle\mathbf{V}_k\rangle_{(V|D)}$. Using partial committee symmetric assumption, we will assume that only one direction is special; thus

$$\lambda\mathbf{W}_k=\langle\mathbf{V}_k\rangle_{(V|D)}+\mu\sum_l\langle\mathbf{V}_l\rangle_{(V|D)},$$

where λ is determined by the normalization condition, whereas μ is a free parameter. It may be shown that the smallest error is obtained for $\mu=0$. The normalized $\mu=0$ solution is $\mathbf{W}_k=\langle\mathbf{V}_k\rangle_{(V|D)}/\sqrt{D+q}$. Inserting this into the generalization function (12), we find to leading order in $1/K$

$$\epsilon_{\mathrm{opt}}(q,d)=\frac{1}{\pi}\arccos\left(\frac{2\left(\frac{d}{\sqrt{q}}+\arcsin\sqrt{q}\right)}{\pi\sqrt{1+\frac{2d}{\pi q}}}\right). \tag{14}$$

D. Small α

In the small $\alpha=O(K^{-1})$ regime the saddle-point solution of the free energy gives $q=0$ [7], i.e., the solution remains symmetric. From this it follows that for optimal learning $(1/N)\mathbf{W}_l\cdot\mathbf{W}_k=1$, i.e., the optimal solution is the simple perceptron. This result shows that when presented with $O(N)$ examples, the best one can do is to be conservative and let the student have only N parameters. The error of the Bayes algorithm (10) and the optimal learning algorithm (14) reduces to the common result $\epsilon_{\mathrm{Bayes}}(d)=\epsilon_{\mathrm{opt}}(d)=(1/\pi)\arccos[\sqrt{(2/\pi)d}]$, whereas the $T=0$ Gibbs error (13) reduces to $\epsilon_{\mathrm{Gibbs}}(d)=(1/\pi)\arccos[(2/\pi)d]$.

Figure 1 shows the Bayes and the optimal learning curve

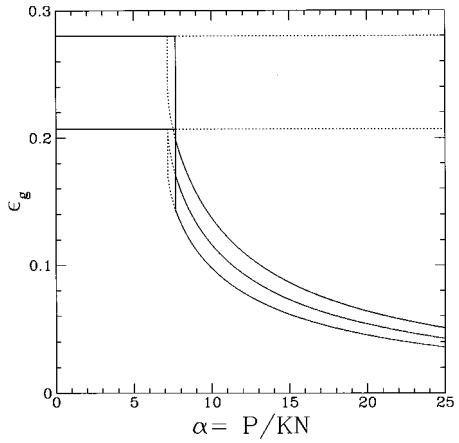

FIG. 2. Learning curve for the committee task in the finite-α regime. The lower full curve is the result for Bayes algorithm. The middle full curve is the result for optimal learning. The upper full curve is the zero temperature Gibbs learning curve. The metastable states are indicated by dotted lines. Below $\alpha=7.68$, the Bayes algorithm and the optimal learning algorithm give identical results.

and the zero-temperature Gibbs learning curve. Asymptotically, i.e., $\alpha K\to\infty$, the typical overlap between two solutions d goes to 1. The Gibbs learning curve shows a strong overfitting effect with an asymptotic error of $(1/\pi)\arccos(2/\pi)=0.28$. The nonzero-temperature Gibbs learning curve has a lower asymptotic error and for $T\to\infty$ it will approach the asymptotic error of Bayes and optimal learning $(1/\pi)\arccos(\sqrt{2/\pi})=0.21$. However, the decay towards the asymptote will be much slower than in the optimal case.

E. Finite α

The specialized solution with $q=O(1)$ will exist for finite $\alpha=O(1)$. In this region the saddle points of the free energy yield the simple relation $d+q=1$ [7]. This leaves us with a free energy as a function of one parameter, say, q. The solution of this saddle-point equation gives the following result. For $\alpha<7.17$ there exists only one solution with $q=0$. This solution corresponds to the residual generalization error of the small-α region shown in Fig. 1. This solution exists for all values of α. At $\alpha=7.17$ an additional specialized $q>0$ solution appears. This solution becomes the minimum of the free energy at $\alpha=7.68$. Therefore, the system makes a first-order transition. The Gibbs learner, the Bayes learner, and the optimal learning curve make a discontinuous drop at $\alpha=7.68$. The generalization error of the different solutions are shown in Fig. 2. The asymptotic behavior of the learning curve for the Bayes algorithm and the optimal learning algorithm may be easily deduced by solving the saddle-point equations in the limit $1-q$ small,

$$\epsilon_{\text{Bayes}}=\epsilon_{\text{opt}}=2\left(\int Dt\frac{e^{-t^2/2}}{H(t)}\right)^{-1}\frac{1}{\alpha}+O\left(\frac{1}{\alpha^2}\right).$$

This result is valid for $K\to\infty$ and $\alpha\ll\sqrt{K}$ and it has a simple relation to the asymptotic learning curve for the Gibbs algorithm obtained by [7]: $\epsilon_{\text{Bayes}}=\epsilon_{\text{opt}}=\epsilon_{\text{Gibbs}}/\sqrt{2}$. This is the same relation between the three algorithms as found for the simple perceptron [5,6], with the prefactor on the error being twice the simple perceptron value. We expect, as it has been observed for the tree committee machine [11], that for any finite K the asymptotic behavior will be the same as for the simple perceptron.

IV. PROTOTYPE TEACHER

In the following we shall study a quite different classification task, the prototype or proximity problem. The Gibbs learning approach to the prototype problem was studied for the simple perceptron in [12] and for the case of the committee machine in [8]. It has also been formulated in the context of optimal learning in [13].

The proximity classifier is characterized by a set M N-dimensional vectors $\mathbf{S}^\mu$ and corresponding binary values $\tau^\mu=\pm1$, which together make up the teacher parameters $V=\{\mathbf{S}^\mu,\tau^\mu\}_{\mu=1}^M$. These parameters are used as *prototype* examples for the classification: For any input vector $\mathbf{S}$ the classifier produces an output $\tau(V,\mathbf{S})$, which is equal to the output of the prototype vector closest to this input. More precisely,

$$\tau(V,\mathbf{S})=\sum_\mu \tau^\mu\prod_{\nu\neq\mu}\Theta(\mathbf{S}\cdot\mathbf{S}^\mu-\mathbf{S}\cdot\mathbf{S}^\nu), \tag{15}$$

where the prototype vectors are normalized $|\mathbf{S}^\mu|^2=N$. We shall assume that the teacher selects the example vectors symmetrically inside a cone around the prototype so that $\mathbf{S}\cdot\mathbf{S}^\mu\leqslant mN$ (with $|\mathbf{S}|^2=N$). In the thermodynamic limit it is straightforward to show that randomly selected examples will lie on the surface of the cone $\mathbf{S}\cdot\mathbf{S}^\mu=mN$ with probability one. Also two examples belonging to the same prototype will have overlap $\mathbf{S}_1\cdot\mathbf{S}_2=m^2N$.

The teacher generates a set of MP training examples, with P^μ belonging to each prototype, so that $\Sigma_\mu P^\mu=MP$. Thus the training set may be denoted by $D=\{\mathbf{S}_p^\mu,\tau^\mu\}$, with $p=1,\ldots,P^\mu$ and $\mu=1,\ldots,M$. For a random selection of examples $P^\mu\approx P$ when $P\gg1$. We will study the learning problem in the same limit as [12] $N\gg P\gg m^2P=O(1)$.

The probability of generating this particular set is, for problems where the order in which the teacher generates the examples is immaterial,

$$\mathcal{P}(D|V)=\prod_{\mu=1}^M\prod_{p=1}^P\mathcal{P}(\mathbf{S}_p^\mu|\mathbf{S}^\mu). \tag{16}$$

The factorization of the probability (16) over the examples $D^\mu=\{\mathbf{S}_p^\mu,\tau^\mu\}_{p=1}^P$ belonging to each prototype is a major simplification. If the *a priori* probability for selecting a prototype factorizes, i.e., if $\mathcal{P}(V)=\Pi_\mu\mathcal{P}(\mathbf{S}^\mu)$, then the total probability that any teacher may produce the data set also factorizes $\mathcal{P}(D)=\Pi_\mu\mathcal{P}(D^\mu)$ into a product of single-cluster probabilities $\mathcal{P}(D^\mu)=\int d\mathbf{S}^\mu\mathcal{P}(\mathbf{S}^\mu)\Pi_{p=1}^P\mathcal{P}(\mathbf{S}_p^\mu|\mathbf{S}^\mu)$. This is therefore also the case for the Bayesian inversion $\mathcal{P}(V|D)=\Pi_\mu\mathcal{P}(S^\mu|D^\mu)$, which will be used below.

A. Bayes algorithm

By means of the definition (15) we find the probability (2) that a new example $\mathbf{S}$ will yield the result σ given the knowledge of the training examples

$$\mathcal{P}(\sigma|D,\mathbf{S})=\sum_{\mu}\Theta(\sigma\tau^{\mu})\left\langle\prod_{\nu\neq\mu}\Theta(\mathbf{S}\cdot\mathbf{S}'^{\mu}-\mathbf{S}\cdot\mathbf{S}'^{\nu})\right\rangle_{(V'|D)}.$$

In calculating this quantity we are using the full Voronoi tessellation because we integrate over all the (primed) prototypes consistent with the examples, while keeping $\mathbf{S}$ fixed. The above formula will, for every set of integrand prototypes, select a unique one that is nearest to $\mathbf{S}$. Integrating over the prototypes, selection may jump around and therefore lead to a nontrivial result.

The complete set of D examples already seen and the new example $\mathbf{S}$ are, however, all generated by a particular choice of prototypes, say, $V=\{\mathbf{S}^{\mu}\}$. This implies (in the thermodynamic limit) that if $\mathbf{S}$ is an example of $\mathbf{S}^{\mu}$ then $\mathbf{S}\cdot\mathbf{S}_p^{\mu}=m^2N+O(1)$ for all the examples of the same prototype and $\mathbf{S}\cdot\mathbf{S}_p^{\nu}=O(\sqrt{N})$ for all other examples (where it is assumed that the prototype vectors are drawn randomly). From this it immediately follows that the optimal Bayes student becomes trivial [except in the extreme case of $m^2=O(1/N)$] and simply outputs the classification of the nearest example (or of the nearest estimator). The Bayes error therefore vanishes for the prototype problem.

B. Optimal learning

Since a new input is chosen at random to belong to one of the prototypes the generalization function becomes the average over the generalization function, for individual subvolumes:

$$\epsilon(W,V)=\langle\Theta(-\sigma(W,\mathbf{S})\tau(V,\mathbf{S}))\rangle_{(\mathbf{S}|V)}$$

$$=\frac{1}{M}\sum_{\mu}\langle\Theta(-\tau^{\mu}\sigma(W,\mathbf{S}))\rangle_{(\mathbf{S}|\mathbf{S}^{\mu})}$$

and likewise for the network error

$$\epsilon_{\mathrm{net}}(W,D)=\frac{1}{M}\sum_{\mu=1}^{M}\langle\Theta(-\tau^{\mu}\sigma(W,\mathbf{S}))\rangle_{(\mathbf{S}|D^{\mu})},\qquad(17)$$

where we used that $\langle f(\mathbf{S})\rangle_{(\mathbf{S}|S^{\mu}),(S^{\mu}|D^{\mu})}=\langle f(\mathbf{S})\rangle_{(\mathbf{S}|D^{\mu})}$. It is straightforward to show that $\mathcal{P}(\mathbf{S}|D^{\mu})\sim\delta(\mathbf{S}\cdot\hat{\mathbf{S}}^{\mu}-N[m^2P/\gamma])$, where

$$\hat{\mathbf{S}}^{\mu}=\frac{1}{\gamma}\sum_{p=1}^{P}\mathbf{S}_p^{\mu}\qquad(18)$$

is the estimator of $\mathbf{S}^{\mu}$ and γ is determined by the normalization $|\hat{\mathbf{S}}^{\mu}|^2=N:\gamma=\sqrt{P+P(P-1)m^2}$. Comparing this with $\mathcal{P}(\mathbf{S}|\mathbf{S}^{\mu})\sim\delta(\mathbf{S}\cdot\mathbf{S}^{\mu}-Nm)$, we see that we can obtain the network error from the generalization function by replacing $\mathbf{S}^{\mu}$ with the estimator $\hat{\mathbf{S}}^{\mu}$ and m by

$$\hat{m}=\frac{m^2P}{\gamma}=m\sqrt{\frac{\hat{P}}{1+\hat{P}}},\quad\hat{P}=\frac{m^2}{1-m^2}P,\qquad(19)$$

which is smaller than m and therefore corresponds to a larger opening angle. The quantity $\hat{P}$ is a conveniently rescaled example count [12]. This result was derived by [13] in a less formal way.

We may now calculate the network error using the result above

$$\epsilon_{\mathrm{net}}(W,D)=\frac{1}{M}\sum_{\mu=1}^{M}\epsilon(\lambda^{\mu},C).\qquad(20)$$

The generalization function is expressed in terms of the standard expression

$$\epsilon(\lambda,C)=\left[\prod_k\int\frac{dy_k d\tilde{y}_k}{2\pi i}\right]\Theta\left(-\frac{1}{\sqrt{K}}\sum_k\mathrm{sgn}y_K\right)$$

$$\times\exp\left(\sum_k\tilde{y}_k(y_k-\lambda_k)+\frac{1}{2}\sum_{k,l}C_{kl}\tilde{y}_k\tilde{y}_l\right),\qquad(21)$$

where the integral over $\tilde{y}_k$ runs along the imaginary axis,

$$\lambda_k^{\mu}=\frac{\hat{m}}{\sqrt{1-\hat{m}^2}}\frac{1}{\sqrt{N}}\tau^{\mu}\hat{\mathbf{S}}^{\mu}\cdot\mathbf{W}_k\qquad(22)$$

is the prototype-weight overlap, and $C_{kl}=(1/N)\mathbf{W}_k\cdot\mathbf{W}_l$ is the usual hidden unit correlation.

What remains to be done to obtain the optimal error (9) is to find the minimum of Eq. (20) with respect to the weights. It is clear that the minimum must be a superposition of prototypes of the form

$$\mathbf{W}_k=\frac{1}{\sqrt{M}}\sum_{\mu}D_k^{\mu}\tau^{\mu}\hat{\mathbf{S}}^{\mu}\qquad(23)$$

with suitable coefficients D_k^{μ}, which may be related to the overlaps through Eq. (22). This result is not very useful because of the extensive number of $MK=O(N)$ order parameters λ_k^{μ}. One could go on to minimize Eq. (20) directly. However, we decide to make simplifying assumptions in order to get a finite set of order parameters. This may not be so unreasonable since for the simple perceptron the Hebb solution $\lambda^{\mu}=\lambda$ has turned out to optimal [13].

We assume that at the minimum of Eq. (20) all prototypes make the same contribution to the error. We will make the same assumption for the correlations of the hidden unit as in Sec. III: $C_{kl}=C+(1-C)\delta_{kl}$. Introducing the hidden states $\sigma_k=\pm1$ we may write

$$\epsilon(\lambda,C)=\sum_{[\sigma]}\Theta\left(-\frac{1}{\sqrt{K}}\sum_k\sigma_k\right)\int Dt\prod_k H\left(\sigma_k\frac{t\sqrt{C}-\lambda_k}{\sqrt{1-C}}\right).\qquad(24)$$

In order to have $\epsilon_{\mathrm{net}}(W,D)=\epsilon(\lambda,C)$ each prototype must have the same set of at most K different overlaps $\{\lambda_k\}$. Since Eq. (24) is invariant under a permutation of the λ_k's we may set $\lambda_k^{\mu}=(\pi^{\mu}\lambda)_k$ where π^{μ} is an arbitrary permutation of the K hidden units.

We shall assume that the permutations are totally random, so that any given overlap λ_a occurs among the MK overlaps with probability $p_a = K_a/K$ in the thermodynamic limit. Using these symmetry assumptions, it follows from the requirement of minimum of Eq. (20) that each weight vector is a linear combination of the prototype sums

$$\hat{\mathbf{S}}_k^a = \frac{1}{\sqrt{M}} \sum_{\mu \in \mathcal{M}_k^a} \tau^\mu \hat{\mathbf{S}}^\mu,$$

where the sum is over the set $\mathcal{M}_k^a$ (of size $|\mathcal{M}_k^a| = p_a M$) of those prototypes that all have overlap λ_a with hidden unit k.

It is easily shown that in the thermodynamic limit we have $(1/N)\hat{\mathbf{S}}_k^a \cdot \hat{\mathbf{S}}_k^b = p_a \delta_{ab} + O(1/\sqrt{N})$. For different hidden units the expression is complicated by the fact that the same prototype may have the same or different overlaps with different hidden units. However, due to the constraint that each prototype should have exactly K_a overlaps of size λ_a, the probability is $K_a K_b/K(K-1)$ for different overlaps and $K_a(K_a-1)/K(K-1)$ if the overlaps are the same. Hence we may write

$$\frac{1}{N} \hat{\mathbf{S}}_k^a \cdot \hat{\mathbf{S}}_l^b = \frac{1}{K-1}(K p_a p_b - p_a \delta_{ab}) + O(1/\sqrt{N}) \quad (k \neq l).$$

It then immediately follows that the expansion of the weight vectors takes the form

$$\mathbf{W}_k = \sqrt{\hat{\alpha}_0} \sum_a \lambda_a \hat{\mathbf{S}}_k^a, \tag{25}$$

collecting all explicit $\hat{P}$ dependences in the parameter

$$\hat{\alpha}_0 = \frac{1+\hat{P}}{\hat{P}} \alpha_0, \quad \alpha_0 = \frac{1-m^2}{m^2} \frac{M}{N}.$$

This solution corresponds to $D_k^\mu = \sqrt{\hat{\alpha}_0}\lambda_a$ for $\mu \in \mathcal{M}_k^a$ in Eq. (23).

For the weight overlaps C_{kl} we get the normalization condition $1 = C_{kk} = \hat{\alpha}_0 \Sigma_a \lambda_a^2 p_a$ and

$$C = C_{kl} = \frac{\hat{\alpha}_0 K \left(\sum_a \lambda_a p_a \right)^2 - 1}{K-1} \quad (k \neq l).$$

Thus the original symmetry *Ansatz* for C_{kl} is verified by Eq. (25). The total number of free parameters may be chosen to be the n overlaps λ_a and the n nonvanishing probabilities p_a subject to the above constraint and of course $\Sigma_a p_a = 1$ (if any p_a vanishes, the number n is effectively reduced by 1). Notice that the λ's determined by minimizing Eq. (24) do not depend on the vectors $\hat{\mathbf{S}}_k^a$.

Finally, we may also calculate the training error (the probability of error on an example in the training set) under these assumptions. We obtain

$$\epsilon_{\text{train}} = \frac{1}{MP} \sum_{\mu,p} \left\langle \Theta\left(-\frac{1}{\sqrt{K}} \sum_k \operatorname{sgn}\left[\frac{1}{\sqrt{N}} \tau^\mu \mathbf{S}_p^\mu \cdot \mathbf{W}_k \right] \right) \right\rangle_{(D|V),V},$$

where the weight vectors are taken from Eq. (25). Writing (for fixed but arbitrary μ and p) $\mathbf{W}_k = \mathbf{W}_k^\mu + \sqrt{\alpha_0/M}\lambda_a \tau^\mu \hat{\mathbf{S}}^\mu$, where $\mu \in \mathcal{M}_k^a$, the integral over $\mathbf{S}_p^\mu$ may be carried out. The result may again be expressed in terms of the standard function (21) $\epsilon_{\text{train}} = \epsilon(\lambda', C)$, where

$$\frac{\lambda_a'}{\lambda_a} = \frac{1+\hat{P}}{\hat{P}} \sqrt{\frac{1+(1-m^2)\hat{P}}{1+\hat{P}}} \approx \frac{1+\hat{P}}{\hat{P}}.$$

1. Total symmetry ($n=1$)

In this case we have $p_1 = 1$ and it follows that $\lambda_1 = 1/\sqrt{\alpha_0}$ and $C = 1$. This is the Hebb solution, which is optimal for the simple perceptron [3]. Thus, in order to exploit the computational powers of the committee machine, it is necessary to break the symmetry further.

2. Broken symmetry ($n=2$)

In this case, there are in general only two free parameters that may be taken to be the overlaps λ_1 and λ_2. For finite K there is a further restriction that Kp_a has to be a nonzero integer. For $K=3$ there is consequently only one possibility, namely, $p_1 = \frac{1}{3}$ and $p_2 = \frac{2}{3}$, and this effectively reduces the number of free parameters to 1. For large K one may simplify Eq. (24) using the central limit theorem to carry out the sum over internal states (see, for example, [8]). This approximation was used for all higher values of K.

In Fig. 3 the behavior of training error and optimal error is depicted as a function of $\hat{P}$. The minimization over the two parameters that are free in this case has been carried out numerically. For small $\hat{P}$, we find $C \simeq 1$ for all values of K, i.e., the optimal solution is close to being the simple perceptron. C decreases with increasing training set size. For large K the asymptotic value of the error behave like $\epsilon = \exp(-aK)/\sqrt{aK}$, with $C = -1/(K-1)$ and $p_1 = 0.092 = 1 - p_2$.

3. Broken symmetry ($n>2$)

For $n = K = 3$ we have also investigated the optimal solution numerically and find that it always degenerates into a solution in which two λ_a's coincide, i.e., the $n=2$ solution. Since all λ_a's must be different, we conclude that there is no

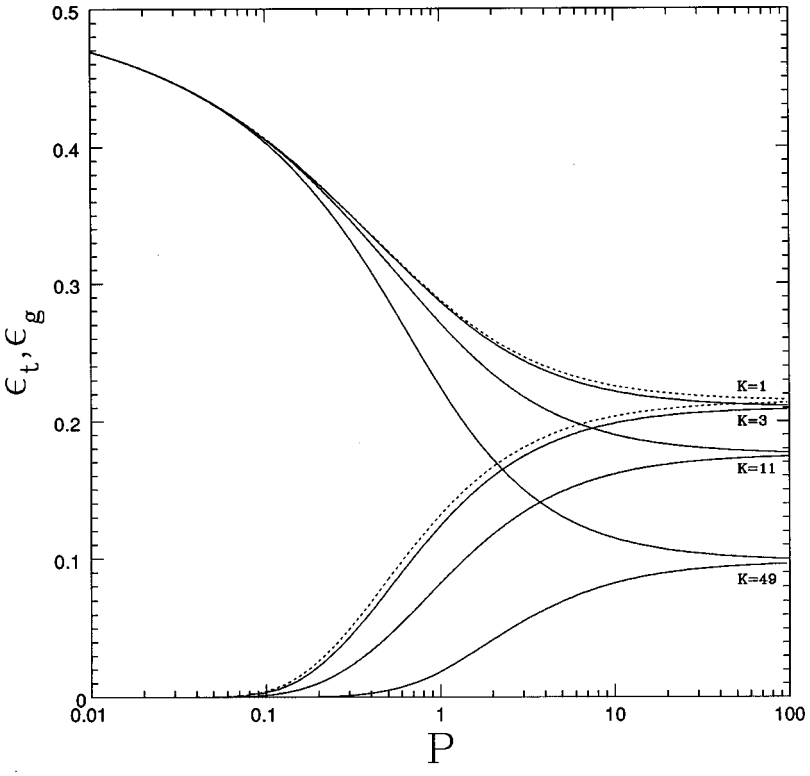

FIG. 3. Optimal learning curve for the proximity problem. The generalization and training error as a function of $\tilde{P}$ for different numbers of hidden units: $K=1$, 3, 11, and 49. $\alpha_0=1.6$.

optimal committee with three members forming three parties for $K=3$. It does not necessarily mean that solutions with $n>2$ for $K>3$ do not exist, but we have not investigated this.

V. DISCUSSION

In this article we have studied two optimal learning algorithms: the Bayes algorithm and the optimal learning algorithm, both of which employ prior knowledge about the problem to be learnt. In the Bayes algorithm the student uses optimal statistics without reference to any specific architecture in order to learn the task presented by the teacher. This algorithm therefore places a lower benchmark for what can be obtained by any other method. In optimal learning, the student is required to have a specific architecture and will make an optimal choice of parameters for this architecture.

We have theoretically studied the performance of these algorithms for a committee machine trained on two classification tasks: the committee machine teacher and the proximity teacher. Ideally, the learning curves we find should provide a lower bound on the generalization error, but in order to find explicit solutions it has been necessary to make certain symmetry assumptions about the order parameters, such as the weight correlations between hidden units.

For committee machine task for training sets of size $O(N)$ (the number of inputs) the solution is committee symmetric with all student weight vectors having the same overlap to all teacher vectors. In the optimal case the solution leads to identical hidden unit vectors (performing together as a simple perceptron), i.e., there is only enough information in the training set to fix N of NK weights. The same generalization error is found for Bayes algorithm in this regime. In the Gibbs case [7] the student vectors, however, are not identical. This leads to a higher generalization error signaling overfitting. The residual generalization error of the symmetric solution is nonzero for both the optimal learning and Gibbs learning algorithms, but higher for Gibbs learning.

For training sets of size $O(NK)$, where K is the number of hidden units, the committee symmetry may be broken and both the Gibbs and optimal learners make a first-order transition to a specialized solution in which the student weight vectors align with their respective teacher vectors. After the transition the decay of the error towards zero is algebraic, being a factor of $\sqrt{2}$ lower for the optimal algorithms asymptotically. We find that in contrast to the simple perceptron and the symmetric phase, Bayes learning is generally better than optimal learning, in spite of the fact that the student and teacher have identical architectures. A committee machine is not the best student of a committee machine. Recently, an algorithm for implementing Bayes algorithm in the tree committee machine has been suggested [14]. In that case it has also been found that Bayes algorithm cannot be implemented by the original teacher architecture.

For the prototype problem the Bayes algorithm gives a trivial result, zero generalization error after presentation of just one example per prototype. The optimal learning of the prototype problem has been studied using the simplest possible symmetry assumption that does not make the network degenerate towards the perceptron. The order parameters of the problem are the embedding strengths of the prototypes. It is assumed that they may take at most a finite set of values. For small training set sizes the best student is close to being a simple perceptron (the correlations between hidden unit are close to one). Increasing the training set, we observe a continuous decrease of the correlations, i.e., a division of labor between the hidden units.

For this problem, optimal learning is easier to study than Gibbs learning, i.e., minimizing the training error, because

we do not have to employ the replica method. For Gibbs learning [8] it has been found that the committee machine needs $O(\sqrt{K})$ examples per prototype to obtain a generalization ability better than the simple perceptron in the limit of $K \to \infty$. It is the fully connected committee machine's ability to store arbitrary patterns that hinder generalization. This shows that using prior knowledge may greatly improve the generalization ability especially in cases when the teacher and student have very different architectures.

ACKNOWLEDGMENTS

The authors are indebted to Holm Schwarze for many discussions and critical comments. J.-B.Z. thanks the Bao Zhao-long and Bao Yu-kang Foundation for financial support. This research is partly supported by the Danish Research Councils for the Natural and Technical Sciences through the Danish Computational Neural Network Center (CONNECT).

[1] J. Hertz, A. Krogh, and R. G. Palmer, *Introduction to the Theory of Neural Computation* (Addison-Wesley, New York, 1991).
[2] E. Gardner and B. Derrida, J. Phys. A **22**, 1983 (1989).
[3] T. L. H. Watkin, A. Rau, and M. Biehl, Rev. Mod. Phys. **65**, 499 (1993).
[4] J. S. Denker, D. Schwartz, B. Wittner, S. A. Solla, R. Howard, L. Jackel, and J. Hopfield, Complex Syst. **1**, 877 (1987).
[5] M. Opper and D. Haussler, Phys. Rev. Lett. **66**, 2677 (1991).
[6] T. L. H. Watkin, Europhys. Lett. **21**, 871 (1993).
[7] H. Schwarze, J. Phys. A **26**, 5781 (1993).
[8] D. O'Kane and O. Winther, Phys. Rev. E **50**, 3201 (1994).
[9] D. Hansel, G. Mato, and C. Meunier, Europhys. Lett. **20**, 471 (1992).
[10] M. Opper, Phys. Rev. Lett. **72**, 2113 (1993).
[11] H. Schwarze and J. Hertz, Europhys. Lett. **20**, 375 (1992).
[12] D. Hansel and H. Sompolinsky, Europhys. Lett. **11**, 687 (1990).
[13] T. L. H. Watkin, K. Y. M. Wong, and A. Rau, in *Proceedings of the International Conference on Artificial Neural Networks, 1993*, edited by S. Gielen and B. Kappen (Springer-Verlag, London, 1993), p. 691.
[14] M. Opper and O. Winther, Phys. Rev. Lett. **76**, 1964 (1996).