On-line versus off-line learning in the linear perceptron: A comparative study

Osame Kinouchi* and Nestor Caticha†
Instituto de Física, Universidade de São Paulo, Caixa Postal 66318, CEP 05389-970 São Paulo, SP, Brazil
(Received 17 March 1995)

The spherical perceptron with N inputs and a linear output does not present optimal generalization if trained by minimization of the standard quadratic cost function

$$\mathcal{E} = \tfrac{1}{2}\sum_{\mu=1}^{\alpha N}(b_\mu - h_\mu)^2,$$

where b_μ and h_μ are the outputs from the rule (teacher) and hypothesis (student) networks for the example μ and there are αN examples. We derive an optimal algorithm for *on-line* learning of examples which outperforms the iterative (*off-line*) standard algorithm for α up to 0.71. The on-line optimized algorithm suggests a class of cost functions for off-line learning, which we then proceed to study using the replica method. The optimized cost function within that class has the suggestive form

$$E = \alpha N\left[\Gamma(1/\alpha N)\sum_{\mu=1}^{\alpha N}[-\ln P(b_\mu \mid h_\mu)] - \Gamma \ln Z\right],$$

where Z is a normalization constant, $P(b_\mu \mid h_\mu)$ is the conditional probability of the output data b_μ given the hypothesis output h_μ, and Γ is a learning parameter analogous to a temperature which decreases in a well defined manner along the learning process.

PACS number(s): 87.10.+e, 02.50.−r, 05.90.+m

I. INTRODUCTION

The ability of generalization is a fundamental cognitivelike property presented by artificial neural networks. Generalization (rule extraction) occurs when the probability of acting successfully on a previously unseen exemplar is larger than just random guessing. This led several authors to study the ability of generalization within the framework of supervised learning [1,2]. The simplest model where this property can be studied systematically is the linear perceptron with no hidden units. This has been done using a statistical dynamical approach [3] and through the use of equilibrium statistical mechanics [4–7].

Given such fixed architecture it seems natural to try to answer questions such as what is the best possible algorithm in the sense of maximizing generalization. The aim of this paper is to study this problem of optimal generalization algorithms, within two possible scenarios of supervised learning. In the first one, we consider the case of *single presentation of examples*, or what has been called *on-line* learning or even *incremental* learning [8–12]. In this case each example is used sequentially, in a manner as prescribed by the learning algorithm, and then thrown away. The synaptic changes made at a given stage of the learning procedure depend specifically only on the example being presented and possibly on the current state of the net. The well known Hebb algorithm [13] for Boolean output perceptrons is a simple instance of on-line learning where all examples receives the same weight independently of the network state. Allowing a modulation mechanism on the Hebbian term has proven to be a very cheap scheme, from a computational cost point of view, for obtaining the same power law decay of the generalization error as standard iterative algorithms [10].

On-line learning is also the natural procedure for time varying rules [11,14] where the examples might not be available all at once, or even when old examples may not be any longer representative of the present state of the rule which has to be inferred. In some cases it might even reduce the "overfitting" effect typical of some iterative methods [15]. It is also the natural scheme for "learning by queries" [8,10], since the criterion for selecting examples depends on the stage of learning. Finally, we observe that on-line learning has been extended to some multilayer networks leading to very interesting results [16–19].

The second scenario of supervised learning to be considered is the so called *off-line* learning. In this case, the synaptic changes depend on the whole set of learning examples defining a global cost function. The examples are used repeatedly until minimization of this cost function is achieved. In this manner the problem of learning has been presented as a problem of equilibrium statisti-

*Electronic address: osame@if.usp.br
†Electronic address: nestor@if.usp.br

cal mechanics [2,4]. All the works on linear perceptrons cited above refer to this off-line scenario. The present study on the optimal linear perceptron complements our previous work on optimal generalization in Boolean perceptrons [10,20].

A further distinction which appears in the case of linear perceptrons is between *constrained* and *unconstrained* learning [3]. In the former case the norm of the perceptron weight vector is kept constant throughout the learning procedure. In the latter, this norm can depend on the learning stage or the number of examples.

For the case of unconstrained learning our results reduce to previous findings [5–7], with perhaps a different point of view on the nature of the cost function. For the case of constrained learning, however, our results are surprising: better generalization is achieved with an *inconsistent algorithm*, that is, without minimizing the *empirical* error (the quadratic difference between the two network outputs). We show that, in this case of constrained learning, minimization of the empirical error leads to overfitting *even if the examples are noiseless*.

In the next section we present the model and discuss the performance measures. In Sec. III, the dynamics of on-line learning is presented and the optimal algorithm determined. The optimized dynamics can be thought of as a gradient descent method which at every time step decreases a cost (*energy*) function which depends explicitly on only the last presented example. This on-line cost function suggests a global cost function, which depends on all the examples of the learning set, to be used in an off-line manner. In Sec. IV the equilibrium statistical mechanics results for this energy function are obtained using the, by now standard, replica method.

The power of single presentation of examples is perhaps best illustrated by the fact that the popular quadratic error energy function with iterative learning, which in practice means substantial computational cost, is outperformed by the computationally cheap optimized on-line algorithm for α up to ≈ 0.71, where α is the number of examples per number of adjustable weights. It is outperformed by the off-line optimized algorithm for α up to 1. For $\alpha > 1$ both iterative methods lead to perfect generalization. Section V presents some concluding remarks concerning our results from a point of view of maximum log-likelihood methods.

II. THE MODEL

A. Learning from examples in perceptrons

The single-layer perceptron output is a function of a weighted sum of N inputs,

$$\sigma_J(\mathbf{S}) = g(h)\,, \qquad h \equiv \sum_j \frac{J_j S_j}{\sqrt{N}} = \sqrt{N}\,\mathbf{J}\cdot\mathbf{S}\,, \tag{1}$$

where h is the perceptron local field, $\mathbf{S} = \{S_j\}$ $(j = 1,\ldots,N)$ is an N-dimensional input vector, $\mathbf{J} = \{J_j\}$ is the perceptron weight vector, and the convention

$$\mathbf{X}\cdot\mathbf{Y} = \frac{1}{N}\sum_j^N X_j Y_j$$

for the scalar product of N-dimensional vectors has been used; $g(x)$ is the perceptron transfer function and it defines the type of machine under study, e.g., we may have a linear perceptron $[g(x) = x]$, a graded response perceptron [$g(x)$ is a sigmoidal function like $\tanh(x)$], or a Boolean perceptron $[g(x) = \mathrm{sgn}(x)]$.

Here we consider the realizable generalization task where the rule to be inferred by the student (or hypothesis) perceptron $\mathbf{J}$ is the map performed by a teacher (or rule) perceptron with unknown weights $\mathbf{B} = \{B_j\}$ but the same architecture and transfer function. Then, the training (or data) set $\mathcal{L} = \{(\mathbf{S}^1,\sigma_B^1),\ldots,(\mathbf{S}^\mu,\sigma_B^\mu),\ldots,(\mathbf{S}^P,\sigma_B^P)\}$ is composed of $P = \alpha N$ input-output pairs $(\mathbf{S}^\mu,\sigma_B^\mu)$ where the input vectors have some probability measure $d\nu_{\mathcal{L}}(\mathbf{S})$ and the desired output is given by

$$\sigma_B^\mu = g(b)\,, \qquad b \equiv \sqrt{N}\,\mathbf{B}\cdot\mathbf{S}^\mu\,, \tag{2}$$

where b is the rule local field.

Usually, the learning process is thought of as an iterative minimization (say, by gradient descent) of some cost function defined by the total data set (off-line learning). This presupposes the storage and repetitive presentation of the learning set (the so called learning epochs). However, a simpler learning process (on-line learning) has been considered where examples are presented only once and sequentially, the change in the perceptron weights being done along the gradient of a cost function defined only by the new example μ and the present network state $\mathbf{J}(\mu-1)$.

B. Performance measures

Different cost functions define learning algorithms with different generalization performances. The generalization performance can be measured through the achieved correlation (or overlap) between the hypothesis and rule vectors, which we write as

$$\rho \equiv \frac{\mathbf{B}\cdot\mathbf{J}}{\sqrt{MQ}} = \cos\theta\,, \tag{3}$$

where $Q \equiv \mathbf{J}\cdot\mathbf{J}$ and $M \equiv \mathbf{B}\cdot\mathbf{B}$ are the student and teacher norms, respectively, and θ is the angle between the two N-dimensional vectors (see Fig. 1). We want to calculate the average value of this correlation $\rho(\alpha)$ as a function of the number of examples per degree of freedom $\alpha = P/N$.

The usual performance measure is the generalization error

$$\begin{aligned} e_g &\equiv \int d\nu_{\mathcal{T}}(\mathbf{S})\frac{1}{2}\left[b(\mathbf{S}) - h(\mathbf{S})\right]^2 \\ &= \frac{1}{2}\left(M + Q - 2\rho\sqrt{MQ}\right)\,, \end{aligned} \tag{4}$$

where $d\nu_{\mathcal{T}}(\mathbf{S})$ is the measure of random input vectors

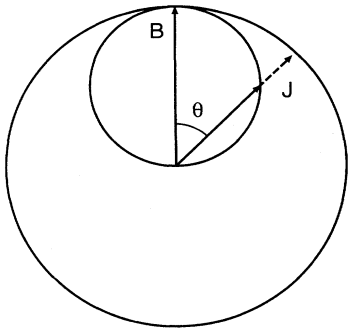

FIG. 1. Geometrical interpretation of the learning problem: the overlap ρ is the cosine of the angle θ between the vectors **B** and **J**. In the constrained case (dashed) the perceptron must lie in the external circle [which represents an $(N-1)$-dimensional spherical surface]. In the unconstrained case (solid) the optimal perceptron is found to lie in the internal surface.

whose components are drawn from a distribution with zero mean and unit variance. We observe that this test set measure $d\nu_{\mathcal{T}}$ may be different from the training set measure $d\nu_{\mathcal{L}}$.

Note that when M and Q are fixed (say, $M = Q = 1$ as in [4]) then maximization of ρ is equivalent to minimization of e_g and we have $e_g = 1 - \rho$. If, however, Q also can be adjusted, then it is trivial to see from (4) that e_g can be further minimized by choosing Q such that $\sqrt{Q} = \rho\sqrt{M}$ (see Fig. 1) giving $e_g = (1-\rho^2)/2$. Since e_g depends not only on the angle between **J** and **B** but also on the modulus of **J** it seems to us that it is not a complete measure for describing the learning process because **J** vectors at different angles from the rule vector **B** will be characterized by the same e_g.

We think that it is important to assign different performance measures to vectors at different angles from the rule. This is the case for Boolean perceptrons where the error measure e_g^B depends univocally on ρ through $e_g^B = \pi^{-1}\arccos\rho$ [2]. We observe that e_g^B can also be used to characterize the linear perceptron: it measures the probability that the hypothesis produces a response with a wrong sign. Perceptrons presenting the same value for e_g may have different values for e_g^B.

We thus prefer to concentrate our attention on the evolution of the overlap $\rho(\alpha)$ as a function of the number of examples, and to consider separately the case with constrained (that is, hypothesis norm Q fixed) and unconstrained learning, as is done by Krogh and Hertz [3].

III. ON-LINE LEARNING

A. The learning equations

A general form for on-line learning procedures can be written as

$$J_i(\mu) = \left(1 - \frac{\Omega_\mu}{N}\right) J_i(\mu-1) + \frac{1}{N} F(\mu) S_i^\mu , \qquad (5)$$

where Ω_μ is a decay parameter (which we may allow to depend on the example μ) and $F(\mu)$ is a function to be variationally determined later. At this point it is worth mentioning that we do not know on what variables it depends, much less its form.

From the corresponding evolution of the scalar products $R(\mu) = \mathbf{B}\cdot\mathbf{J}(\mu)$ and $Q(\mu) = \mathbf{J}(\mu)\cdot\mathbf{J}(\mu)$ we obtain a difference equation (up to order $1/N$) for the evolution of the overlap $\rho(\mu) = R(\mu)/\sqrt{Q(\mu)M}$ [10,14]

$$\rho(\mu) = \rho(\mu-1) + \frac{\rho(\mu)}{N}\left[\left(\frac{1}{\rho(\mu)}\frac{b_\mu}{\sqrt{M}} - \frac{h_\mu}{\sqrt{Q}}\right)\frac{F(\mu)}{\sqrt{Q}} - \frac{I_\mu}{2}\left(\frac{F(\mu)}{\sqrt{Q}}\right)^2\right], \qquad (6)$$

where $h_\mu \equiv \sqrt{N}\,\mathbf{J}(\mu-1)\cdot\mathbf{S}^\mu$ is the hypothesis local field and $I_\mu \equiv \mathbf{S}^\mu\cdot\mathbf{S}^\mu$ is the input vector norm. Note that h_μ is defined by using the *new* example μ with the *previous* state $\mathbf{J}(\mu-1)$.

From Eq. (6) we can obtain, in the limit $N\to\infty$, a differential equation for the overlap evolution in the "continuous" time $\alpha = \mu/N$ [8,10,14],

$$\frac{d\rho}{d\alpha} = \rho\left\langle\left(\frac{1}{\rho}\frac{b_\mu}{\sqrt{M}} - \frac{h_\mu}{\sqrt{Q}}\right)\frac{F(\mu)}{\sqrt{Q}} - \frac{I_\mu}{2}\left(\frac{F(\mu)}{\sqrt{Q}}\right)^2\right\rangle_\mu , \qquad (7)$$

where $\langle\rangle_\mu$ denotes an average over the latest example. The evolution of the norm $\sqrt{Q}$ is obtained by the same procedure, giving

$$\frac{dQ}{d\alpha} = 2Q\left\langle\frac{h_\mu}{\sqrt{Q}}\frac{F(\mu)}{\sqrt{Q}} + \frac{I_\mu}{2}\left(\frac{F(\mu)}{\sqrt{Q}}\right)^2 - \Omega_\mu\right\rangle_\mu . \qquad (8)$$

These equations can be used to determine ρ and Q (and then e_g) for any distribution of examples for any algorithm F. The decay factor Ω_μ can be used to make $dQ/d\alpha = 0$. If we start from the initial condition $Q(0) = 1$ we can mimic the spherical perceptron constraint. If $\Omega_\mu = 0$ we have the case of unconstrained learning.

B. Optimal on-line learning procedure

From a variational analysis applied to Eq. (7) we find that the function $F(\mu)$ which maximizes the overlap increment per example $d\rho/d\alpha$ is

$$F_{opt}(\mu) = \frac{1}{I_\mu}\left[\frac{\sqrt{Q(\mu-1)}}{\rho(\mu-1)}\frac{b_\mu}{\sqrt{M}} - h_\mu\right] . \qquad (9)$$

Without loss of generality the input vector can be normalized such that $I_\mu = 1$; if, however, this is not done the full Eq. (9) should be kept. The optimal learning algorithm then assumes the form

$$J_i(\mu) = \left(1 - \frac{\Omega_\mu}{N}\right) J_i(\mu - 1) + \frac{\sqrt{Q}}{N}\left(\frac{1}{\rho}\frac{b_\mu}{\sqrt{M}} - \frac{h_\mu}{\sqrt{Q}}\right) S_i^\mu . \quad (10)$$

We can define a cost (or energy) function such that the learning dynamics Eq. (5) can be regarded as the instantaneous gradient descent in the space of normalized vectors $\hat{J}_i \equiv J_i/\sqrt{Q}$,

$$\hat{J}_i(\mu) = \left(1 - \frac{\Omega_\mu}{N}\right) \hat{J}_i(\mu - 1) + \frac{1}{N}\frac{\partial E^\mu(\mathbf{J})}{\partial \hat{J}_i} , \quad (11)$$

where $E^\mu(\mathbf{J})$ is the cost function provided by example μ. Since in general we may have E^μ depending on $Q(\mu-1)$, only in the case of constrained learning $Q = \text{const}$ will the derivatives in J_i and $\hat{J}_i$ be equivalent.

We will show below that the training energy which generates the optimal function (9) can be written in the suggestive form [10,14]

$$E^\mu_{opt}(\mathbf{J}) = -\Gamma \ln P(\sigma^\mu_B \mid h_\mu) - \Gamma \ln Z \quad (12)$$

where Z is a constant, Γ is a parameter discussed below, and $P(\sigma^\mu_B \mid h_\mu)$ is the conditional probability of the new output data given the field h_μ [that is, given the present hypothesis $\mathbf{J}(\mu-1)$ and the input $\mathbf{S}^\mu$]. Although irrelevant for the learning process (since it depends only on the derivative of E^μ_{opt}), we conserve the constant term Z for later discussion.

This cost function is reminiscent of maximum log-likelihood methods, but the new element here is the parameter

$$\Gamma = \frac{1-\rho^2}{\rho^2} = \tan^2\theta \quad (13)$$

which can be regarded as a time dependent (or, better, a performance dependent [14]) learning rate parameter which decreases to zero as $\mathbf{J}$ gets closer to $\mathbf{B}$. Although we are in a noiseless learning case, Γ plays the role, in a formal sense, of a temperaturelike quantity and will thus be dubbed the hypothesis temperature, since it is zero if the hypothesis $\mathbf{J}$ is in the "ground state" $\mathbf{B}$ and is infinite if there is no correlation between them. This "temperature," which is a measure of the similarity between rule and hypothesis, should not be confused with the learning temperature usual in the statistical mechanics approach, which describes the noise level of a stochastic learning process.

We now show that the prescription given by Eq. (12) leads indeed to Eq. (10). Consider input vectors whose components are independent identically distributed random variables with zero mean and unit variance. The conditional probability $P(\sigma^\mu_B \mid h_\mu)$ is determined by the law of large numbers and by geometry, and depends only on ρ (which is the cosine of the angle between the vectors $\mathbf{B}$ and $\mathbf{J}$). For linear perceptrons where $\sigma^\mu_B = b_\mu$ we have

$$P(b_\mu \mid h_\mu) = \frac{1}{\sqrt{2\pi(1-\rho^2)}} \times \exp\left(-\frac{(b_\mu/\sqrt{M} - \rho h_\mu/\sqrt{Q})^2}{2(1-\rho^2)}\right) . \quad (14)$$

Introducing this in Eq. (12) we obtain the desired cost function

$$E^\mu_{opt}(\mathbf{J}) = \frac{1}{2}\left(\frac{1}{\rho(\mu)}\frac{b_\mu}{\sqrt{M}} - \frac{h_\mu}{\sqrt{Q}}\right)^2 , \quad (15)$$

if we use (12) with $Z = [2\pi(1-\rho^2)]^{1/2}$. Note that the original dynamics (10) is recovered only after changing from the $\hat{\mathbf{J}}$ to the $\mathbf{J}$ variables, which imply the appearance of a Q term necessary to get the expression for $F(\mu)$, Eq. (9). This energy function naively resembles the standard quadratic error function, but the $1/\rho$ factor has a non-negligible effect which leads to improved performances. We observe that the cost function prescription (12) is also valid for Boolean perceptrons, enabling a unified approach to the generalization problem. In the Boolean case we have $Z = 1$ and

$$E^\mu_{opt} = -\Gamma \ln P(\sigma^\mu_B \mid h_\mu) = -\Gamma \ln H\left(\frac{-\sigma^\mu_B h_\mu}{\sqrt{Q}\sqrt{\Gamma}}\right) ,$$
$$H(x) \equiv \int_x^\infty \frac{dt}{\sqrt{2\pi}} e^{-t^2/2} . \quad (16)$$

This case has been studied in our previous works [10,14,20].

C. Optimal performance for uniform distributions

We consider now the cases where the components of $\mathbf{S}$ are drawn from a uniform distribution in the hypercube $\{\pm 1\}^N$ or from a Gaussian distribution with zero mean and unit variance. We have $\langle b^2\rangle = M$, $\langle h^2\rangle = Q$, and $\langle bh\rangle = \rho\sqrt{MQ}$, leading to the simple differential equations

$$\frac{d\rho}{d\alpha} = \frac{1-\rho^2}{2\rho} , \quad (17)$$
$$\frac{dQ}{d\alpha} = Q\left(\frac{1-\rho^2}{\rho^2}\right) - 2Q\Omega , \quad (18)$$

where we assumed Ω constant. Note that the equation for ρ is now decoupled from the equation for $\sqrt{Q}$ and can be directly integrated leading to

$$\rho_{opt}(\alpha) = \sqrt{1 - e^{-\alpha}} . \quad (19)$$

This is a theoretical upper bound for $\rho(\alpha)$ for any algorithm used in on-line mode. In this form it does not correspond to a practical algorithm because the optimal function F given by Eq. (9) depends on the unknown parameters M and ρ. The form for the optimal weight function suggests that we should estimate (in an on-line manner) these parameters from the data set. A possible way to do this will be discussed in the next section, and for now we will assume the simpler case, usual in the

literature, where M is known.

In the case of constrained learning $Q = M = 1$ the algorithm can be used by substituting in a self-consistent manner the value $\rho(\mu)$ by the theoretical value $\rho(\alpha)$. The optimal function is then

$$F_{opt}^{on\text{-}line} = \frac{b_\mu}{\sqrt{1-e^{-\alpha}}} - h_\mu \,. \tag{20}$$

For unconstrained learning ($\Omega_\mu = 0$) we can write Eq. (18) as

$$\frac{d}{d\alpha}\sqrt{Q} = \frac{\sqrt{Q}}{\rho}\frac{d\rho}{d\alpha} \,. \tag{21}$$

By choosing the *tabula rasa* $Q(0) = 0$ as initial condition, we have $\sqrt{Q} = c\rho$ for some constant c which can be set to 1. In this case the optimal algorithm reduces to $F = \hat{b} - h$, that is, the optimal algorithm for unconstrained learning is equivalent to the standard one (with a prenormalization of the data $\hat{b} = b/\sqrt{M}$).

It is important to stress the fact that the distribution of examples $P(\mathbf{S})$ was used to calculate the corresponding evolution of the overlap $\rho(\alpha)$ but it is not necessary to know it exactly in order to determine the optimal on-line algorithm Eq. (9), which depends on the less specific conditional probability $P(b|h)$. Several distributions $P(\mathbf{S})$ can generate the same conditional distribution $P(b|h)$ with, say, different forms for $P(h)$. In the case of Boolean perceptrons the distribution $P(h)$ affects the evolution of $\rho(\alpha)$ and controlling it may be a good learning strategy (learning by queries [10,14]). It is easy to show from Eq. (7) that, in the case of linear perceptrons, selection of examples leads to no improvement.

D. The standard algorithm

The same calculation can be done for the standard algorithm which uses

$$F(\mu) = b_\mu - h_\mu \,, \tag{22}$$

leading to

$$\frac{d\rho}{d\alpha} = \sqrt{M/Q} - \rho\,\frac{M/Q+1}{2} \,. \tag{23}$$

The specific learning curve depends on the ratio M/Q, supposed constant in the constrained case. The overlap stops increasing ($d\rho/d\alpha = 0$) for the maximum value

$$\rho_{\max} = \frac{2\sqrt{M/Q}}{1+M/Q} \,. \tag{24}$$

The standard algorithm produces $\rho_{\max} = 1$ only when $Q = M$ (see Fig. 2), contrasting with the optimal procedure which achieves $\rho = 1$ even for unrealizable tasks where $Q \neq M$. The corresponding asymptotical values ($\alpha \to \infty$) for the generalization error are

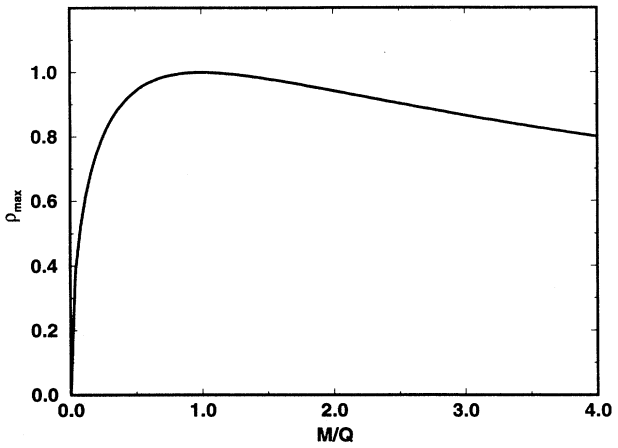

FIG. 2. Asymptotical overlap $\rho_{\max}$ as a function of the ratio M/Q.

$$\begin{aligned} e_g^\infty(\text{standard}) &= \frac{1}{2}\frac{(M-Q)^2}{M+Q} \,, \\ e_g^\infty(\text{optimal}) &= \frac{1}{2}\left(\sqrt{M}-\sqrt{Q}\right)^2 \,, \\ e_g^\infty(\text{standard}) &= \left(1+\frac{2\sqrt{MQ}}{M+Q}\right) e_g^\infty(\text{optimal}). \end{aligned} \tag{25}$$

Even in the realizable case ($Q = M = 1$), we obtain for the standard algorithm the obviously worst result

$$\rho = 1 - e^{-\alpha} = \rho_{opt}^2 \,. \tag{26}$$

By choosing an appropriate decay factor Ω_μ, the norm Q can be held constant, say $Q = M = 1$, enabling a comparison with the results for off-line learning in a spherical perceptron. The equilibrium properties for the standard algorithm $\mathcal{E} = \frac{1}{2}\sum_\mu^{\alpha N}(b_\mu - h_\mu)^2$ have been obtained previously by Seung *et al.* [4]. The overlap achieved after αN examples is

$$\rho(\alpha) = \begin{cases} \alpha & \text{if } \alpha \leq 1 \\ 1 & \text{if } \alpha > 1 \,, \end{cases} \tag{27}$$

which clearly is not optimal since the result for optimal on-line learning Eq. (19) gives, for small α,

$$\rho_{opt} \approx \sqrt{\alpha} + O(\alpha) \,. \tag{28}$$

In Fig. 3 we compare the overlap produced by the various algorithms. It is very interesting that optimal *on-line* learning, with a negligible computational cost, presents better results than *off-line* learning with the standard algorithm up to $\alpha \approx 0.71$.

Concerning specifically on-line learning procedures, the bounds obtained above are also valid for the graded response perceptrons, since in this case the optimal algorithm utilizes the weight function

$$F(\mu) = \sqrt{\frac{Q}{M}}\,\frac{1}{\rho} g^{-1}(\sigma_B^\mu) - h_\mu \,, \tag{29}$$

where $g(x)$ is the perceptron transfer function. Recently,

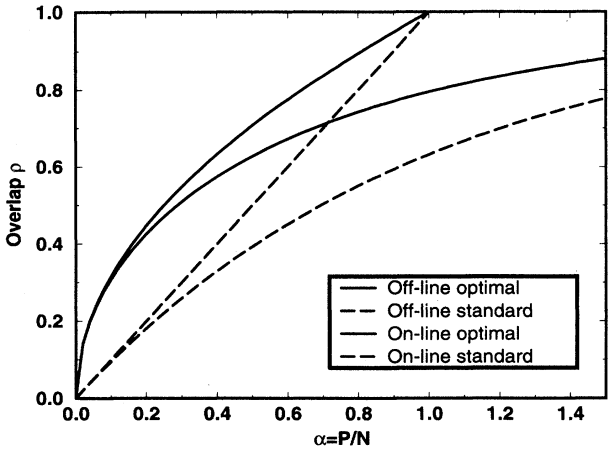

FIG. 3. Overlap $\rho(\alpha)$ versus number of examples for the constrained case: optimal algorithm (solid) and standard algorithm (dashes). Upper curves for off-line and lower curves for on-line learning.

Biehl and Schwarze [12] obtained an asymptotical decay ($\alpha \to \infty$)

$$1 - \rho \propto e^{-0.66\alpha} \tag{30}$$

for on-line learning with the standard quadratic algorithm with an optimized learning step, which must be compared with the lower bound given by Eq. (19),

$$1 - \rho_{opt} \approx \frac{1}{2} e^{-\alpha} . \tag{31}$$

E. Practical algorithms for stationary and drifting rules

By *practical* algorithms we mean procedures which depend only on accessible quantities. Here we examine if the optimal performance can be approximated by some practical learning algorithm. The optimal function F is given by

$$F(\mu) = \frac{b_\mu}{A(\mu)} - h_\mu , \quad A(\mu) \equiv \sqrt{\frac{M}{Q}} \rho(\mu - 1) . \tag{32}$$

For stationary rules we have already observed that the dependence on ρ can be lifted if we use, in the function F, the theoretical value $\rho(\alpha)$ given by Eq. (19) instead of the "true" (and inaccessible) value $\rho(\mu)$. For nonstationary rules we can use the approach used in [14] and approximate the parameter $A(\mu)$ by using Eq. (4),

$$A(\hat{e}_g, Q, M) = \frac{1 + M/Q}{2} - \hat{e}_g/Q , \tag{33}$$

with an on-line estimator for the generalization error

$$\hat{e}_g(\mu) = \left(1 - \frac{\omega_e}{N}\right) \hat{e}_g(\mu - 1) + \frac{\omega_e}{N} \frac{1}{2} (b_\mu - h_\mu)^2 . \tag{34}$$

The constant ω_e is an integration time which, from previous findings [14] and numerical simulations, we suggest to be used with the value $\omega_e \approx 2$ for a good compromise between accumulating enough data for reliable statistics and minimizing the lag from the drifting rule. The norm Q is a measurable (although nonlocal) quantity.

In the case of an unknown teacher length, the dependence on M can be remediated by using a similar on-line estimator

$$\hat{M}(\mu) = \left(1 - \frac{\omega_M}{N}\right) \hat{M}(\mu - 1) + \frac{\omega_M}{N} b_\mu^2 \tag{35}$$

for the true value of M. If we use $\omega_M = 1/\alpha$ we have effectively that $\hat{M}(\mu)$ is an estimate of the teacher length M calculated from the entire learning set. If we use a constant ω_M the average is done only over a time span so that the estimator can be used, with the aid of the previous estimator $\hat{e}_g$, for the tracking of a drifting rule $\mathbf{B}$ which changes not only its direction but also its length.

This use of on-line estimators for unknown learning quantities is by now usual in the literature [14,21,22]. Although somewhat involved, these estimative procedures not only provide a method for using the optimal algorithm but also give a very interesting robustness to the perceptron behavior. There is no separation between training and performance phases, the estimator $\hat{e}_g$ for the generalization error being a vigilance parameter which detects changes in the environment. If $\hat{e}_g$ increases due to changes in the rule, the parameter $A(\hat{e}_g)$ decreases leading to a higher attention paid to the examples as measured by the function F.

Along with the versatility of incremental learning (which naturally forgets old examples no longer representative of the actual environment $\mathbf{B}$) this model provides a simple and analytically solvable example of an artificial neural network behaving as a truly *adaptive* system (and not only as a parametric one). Since it has been demonstrated recently [16,12,18,19] that on-line learning is effective for multilayer machines, we expect that future on-line adaptive multilayer nets constructed from the same principles will substitute for the traditional off-line backpropagation nets for real world problems with changing environments.

IV. OFF-LINE LEARNING

A. The statistical mechanics approach

Now we present a variational calculation for off-line learning with a global energy function defined over the whole learning set that is suggested by the on-line results of the previous section. The statistical mechanics approach [4] considers a stochastic version of the learning algorithm

$$J_i(t+1) = J_i(t) + \frac{1}{N} \frac{\partial V(\mathbf{J})}{\partial J_i} + \frac{1}{N} \frac{\partial E(\mathbf{J})}{\partial J_i} + \frac{1}{\sqrt{N}} \eta_i(t) , \tag{36}$$

where white noise with variance $\langle \eta \cdot \eta \rangle = 2T$ has been added, $E(\mathbf{J})$ is the cost function specific for each algorithm, and $V(\mathbf{J})$ is a potential which assures the spherical normalization. The index t denotes a complete presentation of the learning set (often called a training epoch).

It is well known that this Langevin equation leads, in equilibrium, to a Gibbs distribution for the variables $\{J_i\}$. The partition function is

$$Z_{\mathcal{L}} = \int d\nu(\mathbf{J})\, e^{-\beta E(\mathbf{J})}\,, \tag{37}$$

where $\beta = 1/T$ (T is usually called the learning temperature). The measure $d\nu(\mathbf{J})$ refers to the *a priori* distribution produced by $V(\mathbf{J})$ which, in the case of spherical constraint $\mathbf{J}\cdot\mathbf{J} = Q$, reads

$$d\nu(\mathbf{J}) = \left(\prod_i^N \frac{dJ_i}{\sqrt{2\pi e Q}}\right)\ \delta\left(\sum_i^N J_i^2 - QN\right). \tag{38}$$

The Gibbs distribution is used to perform averages over the *a posteriori* distribution of the weights, the thermal averages denoted by $\langle\,\rangle_T$. Since this distribution, and so the partition function, still depends on the specific realization of the learning set $\mathcal{L}$ (which is a set of random quenched variables), we need to perform a quenched average over the possible learning sets by using the replica method. The formalism of replicas is by now standard [1,2,4] and we present only the results.

Our calculations will be done for a training energy suggested by the optimal on-line algorithm,

$$E(\mathbf{J}, r) = \frac{1}{2}\sum_{\mu=1}^{\alpha N}\left(\frac{\sqrt{Q}}{\sqrt{M}r(\alpha)}b_\mu - h_\mu\right)^2\,, \tag{39}$$

where the parameter $r(\alpha)$, which if our prescription (12) is correct will be found to be the overlap $\rho(\alpha)$, is by now considered an arbitrary function to be optimized at the final stage of our calculations.

The replica formalism is very attractive because the order parameters which appear in the calculations

$$R_a = \langle \mathbf{J}^a \cdot \mathbf{B}\rangle_T, \qquad Q_{ab} = \langle \mathbf{J}^a \cdot \mathbf{J}^b\rangle_T \tag{40}$$

have a natural interpretation in the learning problem. The first is the (non-normalized) average overlap between the hypothesis and the rule perceptrons and the second is the typical overlap between two possible hypotheses or students; a and b are replica indices.

B. Replica symmetric results

If we assume replica symmetry

$$R_a = R, \qquad Q_{ab} = Q\delta_{ab} + q(1-\delta_{ab})\,, \tag{41}$$

which is a valid assumption for the case of noiseless examples [7], we obtain the free energy density as

$$-\beta f = \frac{1}{2}\left[\ln(1-q/Q) + \frac{q - R^2/M}{Q-q} - \alpha\ln\left[1+\beta(Q-q)\right] - \alpha\beta\,\frac{q + Q/r^2 - 2R\sqrt{Q/Mr^2}}{1+\beta(Q-q)}\right]. \tag{42}$$

It is useful to write the free energy as a function of the normalized quantities

$$\rho = \frac{R}{\sqrt{QM}}, \qquad \tilde{q} = \frac{q}{Q}, \qquad \tilde{\beta} = \frac{Q}{T} = \frac{1}{\tilde{T}}, \tag{43}$$

which are the hypothesis-rule overlap, the interhypothesis overlap, and the ratio between the noise variance and the weight vector length, respectively. The relative quantity $\tilde{T}$ is clearly the relevant measure of the "temperature," and not the absolute noise variance itself. The free energy assumes the more transparent form

$$-\tilde{\beta}\frac{f}{Q} = \frac{1}{2}\left[\ln(1-\tilde{q}) + \frac{\tilde{q}-\rho^2}{1-\tilde{q}} - \alpha\ln\left[1+\tilde{\beta}(1-\tilde{q})\right] - \alpha\tilde{\beta}\,\frac{\tilde{q} + 1/r^2 - 2\rho/r}{1+\tilde{\beta}(1-\tilde{q})}\right]. \tag{44}$$

The order parameters ρ and $\tilde{q}$ are given by the saddle point equations $\partial f/\partial\rho = 0$ and $\partial f/\partial\tilde{q} = 0$. After some simple algebra we obtain

$$\rho = \frac{\alpha}{r}\,\frac{x}{1+x}\,, \tag{45}$$

$$\tilde{q} = \frac{\alpha}{r^2}\,\frac{x^2}{(1+x)^2 - \alpha x^2}\left[1 - \alpha\frac{(x-1)}{(x+1)}\right], \tag{46}$$

with the shorthand $x = \tilde{\beta}(1-\tilde{q})$.

In the limit of zero learning temperature ($x \to \infty$) we obtain simply

$$\rho = \frac{\alpha}{r}, \qquad \tilde{q} = \frac{\alpha}{r^2}\,. \tag{47}$$

C. The effect of the parameter $r(\alpha)$

The correlation between hypotheses depends on the function $r(\alpha)$ used in the algorithm, and $\rho(\alpha)$ can be optimized if we choose the minimal $r(\alpha)$ possible. Since the maximum value for the overlap between the hypotheses is $\tilde{q} = 1$, we find

$$r_{opt}(\alpha) = \sqrt{\alpha}\,, \tag{48}$$

which leads to

$$\rho_{opt}(\alpha) = \sqrt{\alpha}\,. \tag{49}$$

It is important to observe that the same result may be obtained if we extremize f with respect to r. The condition $\partial f/\partial r = 0$ gives directly $r_{opt} = \rho$.

The result $r_{opt}(\alpha) = \rho_{opt}(\alpha)$ corroborates our prescription (12) derived from the optimal on-line algorithm. We

conjecture that this prescription leads, indeed, to an optimal off-line algorithm, although this has not been rigorously proven. For now, we will call this algorithm the "optimal" one (within quotes). The value $\tilde{q} = 1$ means that there is only one "optimal" generalization vector which is determined by the learning set. Seung *et al.* [4], in contrast, obtain that $\tilde{q} = \alpha M/Q$ for the overlap between the hypotheses produced by the standard algorithm (which is obtained using $r = \sqrt{Q/M}$). We observe that all these results are valid for $\alpha < 1$. For $\alpha > 1$ we have perfect generalization $\rho = 1$.

It is curious that in both the off-line and on-line cases the simple relation $\rho_{opt} = \sqrt{\rho}$ emerges, where ρ_{opt} is the "optimal" algorithm result, while ρ is the result obtained by using the empirical error as a cost function. A similar relation $\rho_{Bayes} = \sqrt{\rho}$ occurs in the case of a Boolean perceptron [23,20] between the optimal "Bayes" overlap and that obtained by the "Boltzmann" algorithm for $T = 0$. The Boltzmann algorithm also minimizes the empirical error

$$\mathcal{E} = \sum_{\mu=1}^{\alpha N} \Theta(-\sigma_B^\mu \sigma_J^\mu) = \frac{1}{4} \sum_{\mu=1}^{\alpha N} (\sigma_B^\mu - \sigma_J^\mu)^2 , \tag{50}$$

being very similar to the standard algorithm for the linear case. Both are called consistent algorithms in the literature because they produce zero error in the training set.

It is important to note that the optimized algorithm does not minimize the empirical error. It is possible to show by standard methods that its average empirical error is

$$e_t \equiv \frac{1}{\alpha N} \langle \langle \mathcal{E}_T \rangle_{\mathcal{L}} = \frac{1}{2} \left(\sqrt{M} - \frac{\sqrt{Q}}{\rho} \right)^2 , \tag{51}$$

which is clearly nonzero even for noiseless examples and zero temperature (it is an inconsistent algorithm).

Summarizing, for the realizable case $Q = M = 1$, we have

$$e_g(\text{opt}) = 1 - \sqrt{\alpha} , \tag{52}$$

$$e_t(\text{opt}) = \frac{1}{2\alpha} (1 - \sqrt{\alpha})^2 , \tag{53}$$

$$e_g(\text{std}) = 1 - \alpha , \tag{54}$$

$$e_t(\text{std}) = 0 , \tag{55}$$

which means that $e_t(\text{std}) < e_t(\text{opt})$ but $e_g(\text{opt}) < e_g(\text{std})$.

D. Unconstrained learning

In the case of unconstrained learning [3,5,7] there is an obvious choice for the perceptron norm which minimizes the distance $\mathbf{D} = \mathbf{B} - \mathbf{J}$. This condition is represented in Fig. 1 by the internal circle, where $\sqrt{Q(\alpha)} = \sqrt{M} \cos\theta = \sqrt{M}\rho$. In this case (but only in this case) the "optimal" algorithm Eq. (39) coincides with the standard one, producing the so called pseudoinverse solution which has the same $\rho = \sqrt{\alpha}$ behavior. From Eq. (51) we see that the pseudoinverse has zero training error.

We observe that in the case of constrained learning the solution vector (which we have found to be unique, $\tilde{q} = 1$) lies in the direction of the pseudoinverse vector, differing only in its length. It is important to note that the pseudoinverse solution is *not* a solution for the realizable constrained case where the hypothesis space contains only vectors with $Q = M$.

V. CONCLUSIONS

We have used a variational approach to study learning in the linear perceptron. The advantage of this method is twofold: it gives an upper bound for the performance of learning algorithms in various learning situations; it also gives an ideal F_{opt} modulation function which may be approximated by practical learning algorithms. We thus presented a comparison between the standard algorithm and the optimal one found by the variational procedure for various scenarios: on-line and off-line learning with constrained or unconstrained hypothesis spaces.

While for the unconstrained case the standard algorithm is equivalent to the optimal one, this is not true when Q is constrained to be, say, equal to M. In this last case it holds that $\rho_{opt} = \sqrt{\rho_{std}}$ for both on-line and off-line learning.

The on-line optimal curve $\rho_{opt}^{on\text{-}line} = \sqrt{1 - \exp(-\alpha)}$ is an upper bound for any on-line algorithm used to train a linear perceptron as well as any other single-layer machine with a continuous monotonic transfer function.

The off-line optimal performance $\rho_{opt}^{off\text{-}line} = \sqrt{\alpha}$ is the same as that produced by the pseudoinverse vector, which solves the unconstrained case. The standard algorithm has zero training error and is thus called consistent. It has a nonunique ground state solution ($\tilde{q} = \alpha$). The optimal constrained vector is unique ($\tilde{q} = 1$) and is an inconsistent algorithm (its training error is always nonzero).

The off-line optimal algorithm suggested by the optimal on-line procedure is equivalent to a synaptic dynamics that minimizes the cost function

$$E(\mathbf{J},t)/Q = \alpha N \left[\Gamma \frac{1}{\alpha N} \sum_{\mu=1}^{\alpha N} [-\ln P(b_\mu \mid h_\mu)] - \Gamma \ln Z \right] = \frac{1}{2} \sum_{\mu=1}^{\alpha N} \left(\frac{b_\mu}{\rho(t)\sqrt{M}} - \frac{h_\mu}{\sqrt{Q}} \right)^2 , \tag{56}$$

where $\Gamma(t) = [1 - \rho^2(t)]/\rho^2(t)$. We can estimate $\rho(t)$ by some method or, if we are interested only in the equilibrium performance (without caring about learning times), we can use the equilibrium value $\rho(t \to \infty) = \sqrt{\alpha}$ as has been done in the previous section.

This can be regarded as a maximum log-likelihood

method with an optimally decreasing learning rate $\Gamma(t)$. But if we consider the function $E(\mathbf{J},t)$ as the *true* cost function [that is, $\Gamma(t)$ being an essential part of the algorithm], then this cost function is more an energylike than an entropylike quantity as usually regarded in the optimization literature.

The parameter Γ has a non-negligible effect in the on-line case. However, since the effect of a learning step vanishes for the equilibrium properties when minimizing a cost function, the above considerations (entropy versus energy interpretation) would seem to be irrelevant for the off-line case. An interesting possibility suggested by the above results is, however, that even for iterative learning, the step parameter has a nontrivial effect on the learning times.

We conjecture that the minimal characteristic learning time (a lower bound) will be obtained by using a learning rate given by $\Gamma(t)$. This could be checked by using the dynamical approach of Krogh and Hertz [3]. If this is true, these results suggest rethinking maximum-likelihood methods by incorporating the learning rate parameter (the "hypothesis temperature") as an essential and nontrivial component of the cost function E which has "energy" (and not "entropy") as its physical analogue.

Finally, we stress the fact that this constrained learning scenario provides a simple and clear example where the naive minimization of the empirical error leads to overfitting even with a realizable task (the rule pertains to the hypothesis space) and noise-free data. The "optimal" algorithm is an inconsistent one, but has better generalization than consistent algorithms. If the data have insufficient information for determining the rule ($\alpha < 1$) it is not a good strategy to choose the best-fitting model. Since the prediction is imperfect, a perfect hindsight signals that the model is biased to those particular past data.

We have also shown that the proper choice of the cost function leads to an on-line performance better than the brute force off-line learning with the standard algorithm for $\alpha < 0.71$. These results are consistent with previous findings on the generalization properties of Boolean perceptrons [10,20]. We are presently extending this approach to multilayer networks [19].

ACKNOWLEDGMENTS

O.K. and N.C. received support from the Brazilian agencies CNPq and FAPESP.

[1] J. A. Hertz, A. Krogh, and R. G. Palmer, *Introduction to the Theory of Neural Computation* (Addison-Wesley, Redwood City, CA, 1991).
[2] T. L. H. Watkin, A. Rau, and M. Biehl, Rev. Mod. Phys. **65**, 499 (1993).
[3] A. Krogh and J. A. Hertz, J. Phys. A **25**, 1135 (1992).
[4] S. Seung, H. Sompolinsky, and N. Tishby, Phys. Rev. A **45**, 6056 (1992).
[5] S. Bös, W. Kinzel, and M. Opper, Phys. Rev. E **47**, 1384 (1993).
[6] A. P. Dunmur and D. J. Wallace, J. Phys. A **26**, 5767 (1993).
[7] J. F. Fontanari, J. Phys. A **26**, 6147 (1993).
[8] W. Kinzel and P. Rujan, Europhys. Lett. **13**, 473 (1990).
[9] O. Kinouchi and N. Caticha, Physica A **185**, 411 (1992).
[10] O. Kinouchi and N. Caticha, J. Phys. A **25**, 6243 (1992).
[11] M. Biehl and H. Schwarze, J. Phys. A **26**, 2651 (1993).
[12] M. Biehl and H. Schwarze, J. Phys. A **28**, 643 (1995).
[13] F. Vallet and J.-G. Cailton, Phys. Rev. A **41**, 3059 (1990).
[14] O. Kinouchi and N. Caticha, J. Phys. A **26**, 6161 (1993).
[15] O. Kinouchi and N. Caticha (unpublished).
[16] Y. Kabashima, J. Phys. A **27**, 1917 (1994).
[17] M. Biehl and P. Riegler, Europhys. Lett. **28**, 525 (1994).
[18] M. Copelli and N. Caticha, J. Phys. A **28**, 1615 (1995).
[19] M. Copelli, O. Kinouchi, and N. Caticha (unpublished).
[20] O. Kinouchi and N. Caticha (unpublished).
[21] M. Biehl, P. Riegler, and M. Stechert, Würzburg University, Report No. WUE-ITP-95-007, 1995 (unpublished).
[22] N. Barkai, H. S. Seung, and H. Sompolinsky (unpublished).
[23] M. Opper and D. Haussler, Phys. Rev. Lett. **66**, 2677 (1991).