Information theory and local learning rules in a self-organizing network of Ising spins

Michael Haft, Martin Schlang, and Gustavo Deco
Corporate Research and Development, ZFE T SN 4, Siemens AG, 81730 Munich, Germany

(Received 26 August 1994)

The Boltzmann machine uses the relative entropy as a cost function to fit the Boltzmann distribution to a fixed given distribution. Instead of the relative entropy, we use the mutual information between input and output units to define an unsupervised analogy to the conventional Boltzmann machine. Our network of Ising spins is fed by an external field via the input units. The output units should self-organize to form an "internal" representation of the "environmental" input, thereby compressing the data and extracting relevant features. The mutual information and its gradient with respect to the weights principally require nonlocal information, e.g., in the form of multipoint correlation functions. Hence the exact gradient can hardly be boiled down to a local learning rule. Conversely, by using only local terms and two-point interactions, the entropy of the output layer cannot be ensured to reach the maximum possible entropy for a fixed number of output neurons. Some redundancy may remain in the representation of the data at the output. We account for this limitation from the very beginning by reformulating the cost function correspondingly. From this cost function, local Hebb-like learning rules can be derived. Some experiments with these local learning rules are presented.

PACS number(s): 87.10.+e, 42.79.Ta, 89.70.+c, 05.20.−y

I. INTRODUCTION

The human brain is able to extract important features from the enormous flow of data supplied by its sensory systems. Which features are important depends on the special environment the individual has to survive in. Therefore it is of obvious advantage for every individual to be able to adapt its information processing to a changing environment. The hypothesis may be set up that this process is based mainly on efficient unsupervised learning mechanisms. We denote this form of unsupervised learning as "environment driven self-organization."

Let us consider an example to illustrate these catchwords. While we easily recognize words and names in our own language, a foreign language often sounds strange to us, and it is difficult to discern words and remember the names of persons. Fine nuances in pronunciation of words might correspond to different meanings but are inaudible to a nonskilled person. Typically, it takes a while to get accustomed to the sound of a foreign language. This adaptation process might correspond to the extraction of new relevant features which are useful to discriminate and recognize words in the foreign language. This feature extraction may be viewed as an unsupervised process driven by listening to the sound of the foreign language, that is, it is driven by the special environment we are living in. The capability of recognition of words, names, and faces can be attributed to higher areas of the cortex. Hence this example suggests that even at higher levels of information processing unsupervised learning might be relevant.

Information theory provides us with efficient tools to get better insight into unsupervised learning mechanisms. In recent work, several authors developed learning rules on the basis of information theory. Linsker [1] introduced the "infomax" principle, which proposes the mutual information between input and output units as a criterion for the performance of the network. For linear neurons without noise and a Gaussian input distribution this infomax principle is closely related to a principal component analysis (PCA) [2]. A PCA can be achieved with Hebbian learning rules in a network of linear neurons with lateral connections of the output neurons. Different network topologies have been investigated [3–5]. The lateral weights serve to decorrelate output neurons with an anti-Hebbian rule [6].

Most of these results are deduced by assuming that the input distribution is Gaussian. For this special case local learning rules are shown to be sufficient to maximize an information theoretic measure even in the noise case [2]. A Gaussian distribution is a maximum entropy distribution characterized by the variance in the different directions only. Hence it should not be astonishing that for a distribution with these characteristics only and linear neurons the infomax principle results in extracting the principal components. For general input distributions this is not the case. An information processing system should be able to treat more complicated statistical dependences.

Little is known about unsupervised information processing with nonlinear neurons [7–9]. Most commonly these papers are based on the (somewhat troublesome) concept of the continuous entropy. There are essential differences from the concept of the discrete entropy, even in the limit of an infinite discretization of a continuous variable. At the heart of these differences we may view the mathematical model of the real axis consisting of an uncountable-infinite number of elements. We may attribute different meanings to any of these elements. This may lead to infinite expressions, even for the mutual information, if the resolution of the real axis is not rendered finite by the assumption of noise.

1063-651X/95/52(3)/2860(12)/$06.00

The discrete entropy does not have these problems. In the discrete case, we have only a finite number of distinguishable states. The binary variables of the Ising spins used here may be considered as $\in\{0,1\}$, as well as $\in\{\pm 1\}$ or any other mathematical useful coding. We emphasize this to prevent confusion with other work, e.g., [8], where a continuous sigmoid-shaped neuron is used with output values $\in[0,1]$. While our network maps the input onto a binary code word of length N, the neuron in [8] maps the input onto a real number.

In [10], we find an information-theoretic ansatz with binary variables. Here many interacting spins form a network similar to a Boltzmann machine with input and output units. However, instead of the relative entropy in the conventional Boltzmann machine the mutual information between input and output is used as a cost function [11].

Why is it desirable to have high mutual information? The answer is the standard interpretation of the entropy as a measure of uncertainty. High mutual information means low uncertainty, on average, about the input if we know the evoked output. The mutual information is maximal if we have a bijective mapping of the "environmental" input onto an "internal" representation, the output of our network. For the above given example of recognition of words and names bijectively means we are able to understand different words and discriminate the names of different persons. We may say we are well informed about our environment. Hence infomax means that the output units should self-organize to form an internal representation of the environment, which optimally results in a bijective mapping. For a low-dimensional output this corresponds to a compression of the input data. To get there the network is enforced to extract the main characteristic features of the input and find statistical dependences that can be used in order to build up a more efficient representation of the environment with less redundancy.

The aim of the present work is to investigate possibilities and limitations of local learning rules with respect to the above-defined task. Thereby we will make no assumptions about the input probability distribution. We will see that the entropy is a global measure in the sense that it depends on nonlocal expressions. Hence, when restricted to local learning rules and correspondingly to the conventional two-point interaction of the Ising spins, some redundancy may remain in the representation of the data at the output. We will account for these limitations from the very beginning by rewriting the cost function correspondingly. This new cost function neglects nonlocal multipoint correlations, which may lead to a representation of the data at the output that is not of maximum efficiency. However, the gradient of this cost function will lead to a local learning rule. Local learning rules are very simple, fast to implement, and of biological relevance. However, we do not intend to explain biological information processing; rather we are interested in contributing to the general question of local learning rules in a self-organizing network within the framework of information theory. Nevertheless, we have added one biologically motivated experiment, the retina model (Sec. VI), which we think is generally instructive.

The paper is organized as follows. In Sec. II we define our notation and review the way to the exact learning rule corresponding to the gradient of the mutual information [11]. In Sec. III we use some very rough approximations to sketch the main features of the exact learning rule, the close connection to the Hebbian principle and the origin of anti-Hebbian decorrelation. Section IV introduces the mean field approximation and shows the general limitations of local learning rules. A cost function complying with these limitations is presented. Section V is concerned with the evaluation of the gradient in this cost function. Some experiments with a corresponding local learning rule are performed in Sec. VI. We close by summarizing our main results and by presenting an outlook on future work in Sec. VII.

II. THE EXACT LEARNING RULE

The topology of the network we will use in the following is shown in Fig. 1. The state of the input units is denoted by the vector $\vec{\gamma}$, the output units by $\vec{\alpha}\in\{-1,+1\}^N$, which are termed internal units. $P_{\vec{\gamma}}$ is the probability distribution of the input vectors. This distribution is assumed to be fixed and can be interpreted as the environment acting on our network. During the presentation of the input pattern $\vec{\gamma}$ the input units are clamped in the state $\vec{\gamma}$. Via the matrix of "feedforward" weights F, the field $F\vec{\gamma}$ acts as an external field on the $\vec{\alpha}$ neurons. $F\vec{\gamma}$ is real valued; hence even $\vec{\gamma}$ may be continuous: $\vec{\gamma}\in\mathbb{R}^K$. One row of F is denoted by $\vec{f}_i$. W is the symmetric matrix of "recurrent weights" interconnecting the $\vec{\alpha}$ neurons.

For a given $\vec{\gamma}$ the energy function $H_{\vec{\gamma}}(\vec{\alpha})$ of this Ising spin system is

$$H_{\vec{\gamma}}(\vec{\alpha})=-\vec{\alpha}^T F\vec{\gamma}-\tfrac{1}{2}\vec{\alpha}^T W\vec{\alpha}\ . \tag{1}$$

Thus the partition function $Z_{\vec{\gamma}}$ and the conditional probability $P_{\vec{\alpha}/\vec{\gamma}}$ for $\vec{\alpha}$ given a fixed $\vec{\gamma}$ are

$$Z_{\vec{\gamma}}=\sum_{\alpha}\exp[-H_{\vec{\gamma}}(\vec{\alpha})] \tag{2}$$

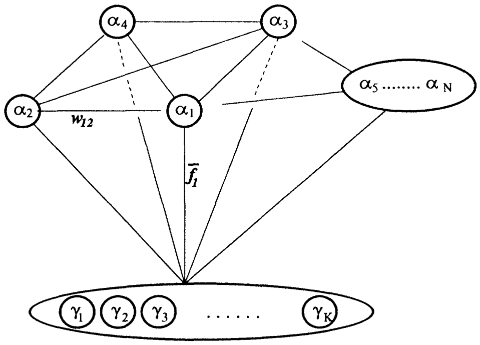

FIG. 1. Notation and topology of the network described herein.

and

$$P_{\vec{\alpha}/\vec{\gamma}}=\frac{1}{Z_{\vec{\gamma}}}\exp[-H_{\vec{\gamma}}(\vec{\alpha})]\ . \tag{3}$$

We absorbed the temperature as a scaling factor into the weights.

The mutual information $M[\vec{\alpha};\vec{\gamma}]$ between input $\vec{\gamma}$ and output $\vec{\alpha}$ is given by

$$M[\vec{\alpha};\vec{\gamma}]=\sum_{\alpha,\gamma}P_{\vec{\alpha}\vec{\gamma}}\ln\frac{P_{\vec{\alpha}/\vec{\gamma}}}{P_{\vec{\alpha}}}=\left\langle\ln\frac{P_{\vec{\alpha}/\vec{\gamma}}}{P_{\vec{\alpha}}}\right\rangle\ . \tag{4}$$

The weights of the network now have to be adjusted in such a way that the information that the internal units $\vec{\alpha}$ carry about the input $\vec{\gamma}$ is maximized. To find a corresponding learning rule, we evaluate the gradient of the mutual information with respect to the weights F and W. Using (1)–(3) we find for a weight w_{ij} connecting the two output neurons α_i and α_j

$$\frac{\partial}{\partial w_{ij}}P_{\vec{\alpha}/\vec{\gamma}}=P_{\vec{\alpha}/\vec{\gamma}}[\langle\alpha_i\alpha_j\rangle^{\vec{\alpha}\vec{\gamma}}-\langle\alpha_i\alpha_j\rangle^{\vec{\gamma}}]\ , \tag{5}$$

where $\langle\ \rangle^{\sigma}$ denotes the average for fixed σ over all other degrees of freedom. For example, for a function g that depends on two variables β and σ we define $\langle g(\beta,\sigma)\rangle^{\sigma}\equiv\sum_{\beta}P_{\beta/\sigma}g(\beta,\sigma)$. For the weight f_{ij} connecting the input neuron γ_j with the output neuron α_i, everything is analogous. Equation (5), after some algebra, leads to the "exact" learning rule

$$\begin{aligned}\Delta f_{ij}&=\eta\frac{\partial}{\partial f_{ij}}M[\vec{\alpha};\vec{\gamma}]\\&=\eta\left\langle\ln\left[\frac{P_{\vec{\alpha}/\vec{\gamma}}}{P_{\vec{\alpha}}}\right][\langle\alpha_i\gamma_i\rangle^{\vec{\alpha}\vec{\gamma}}-\langle\alpha_i\gamma_j\rangle^{\vec{\gamma}}]\right\rangle\ ,\\\Delta w_{ij}&=\eta\frac{\partial}{\partial w_{ij}}M[\vec{\alpha};\vec{\gamma}]\\&=\eta\left\langle\ln\left[\frac{P_{\vec{\alpha}/\vec{\gamma}}}{P_{\vec{\alpha}}}\right][\langle\alpha_i\alpha_j\rangle^{\vec{\alpha}\vec{\gamma}}-\langle\alpha_i\alpha_j\rangle^{\vec{\gamma}}\right\rangle\ .\end{aligned} \tag{6}$$

III. MAIN FEATURES OF THE EXACT LEARNING RULE

It is computationally very expensive to implement Eq. (6) as a learning rule. The probability distribution $P_{\vec{\gamma}}$ must be known and $P_{\vec{\alpha}/\vec{\gamma}}$ must be calculated for all $\vec{\alpha}\in\{-1,+1\}^N$ in every learning epoch. Some experiments with only a few neurons and a small number of input patterns are done in [11]. Of course, Eq. (6) is nonlocal.

We may approximate the exact learning rule as long as we are sure that the approximation has a positive projection onto the exact gradient. To clarify the main features of (6) we will make some very rough approximations without asking for their validity. For example, let us expand the logarithm in (6):

$$\ln\frac{P_{\vec{\alpha}/\vec{\gamma}}}{P_{\vec{\alpha}}}\approx1-\frac{P_{\vec{\alpha}}}{P_{\vec{\alpha}/\vec{\gamma}}}\ . \tag{7}$$

For the weight f_{ij} this results in

$$\frac{\partial}{\partial f_{ij}}M\approx\langle\alpha_i\gamma_j\rangle-\langle\alpha_i\rangle\langle\gamma_j\rangle\ , \tag{8}$$

which is the Hebb rule apart from mean values. This shows the close connection between mutual information and Hebbian learning.

For the recurrent weights w_{ij} the expansion of the logarithm in Eq. (6) results in $\Delta w_{ij}=0$. Without corrections of the recurrent weights, the output neurons correspond to independent neurons. (We could have started with $W=0$.) There is no local learning rule without recurrent weights that does anything useful. All neurons will work independently of each other and extract the most important feature of the input, thereby losing other features. If two output units α_1 and α_2 do the same, the mutual information $M[\alpha_1;\alpha_2]$ between these two neurons is maximal. Maximizing the mutual information $M[\alpha_1\alpha_2;\vec{\gamma}]$ between input $\vec{\gamma}$ and a two-dimensional output $(\alpha_1\alpha_2)$ automatically includes the minimization of $M[\alpha_1;\alpha_2]$. Without noise α_1 and α_2 should represent mutually independent information. We can see this explicitly by splitting $M[\alpha_1\alpha_2;\vec{\gamma}]$ into four terms:

$$\begin{aligned}M[\alpha_1\alpha_2;\vec{\gamma}]&=\left\langle\ln\left[\frac{P_{\alpha_1\alpha_2/\vec{\gamma}}}{P_{\alpha_1\alpha_2}}\frac{P_{\alpha_1\vec{\gamma}}P_{\alpha_2\vec{\gamma}}P_{\alpha_1}P_{\alpha_2}}{P_{\alpha_1/\vec{\gamma}}P_{\alpha_2/\vec{\gamma}}P^2_{\vec{\gamma}}P_{\alpha_1}P_{\alpha_2}}\right]\right\rangle\\&=\left\langle\ln\frac{P_{\alpha_1\vec{\gamma}}}{P_{\alpha_1}P_{\vec{\gamma}}}\right\rangle+\left\langle\ln\frac{P_{\alpha_2\vec{\gamma}}}{P_{\alpha_2}P_{\vec{\gamma}}}\right\rangle-\left\langle\ln\frac{P_{\alpha_1\alpha_2}}{P_{\alpha_1}P_{\alpha_2}}\right\rangle\\&\quad+\left\langle\ln\frac{P_{\alpha_1\alpha_2/\vec{\gamma}}}{P_{\alpha_1/\vec{\gamma}}P_{\alpha_2/\vec{\gamma}}}\right\rangle\\&=M[\alpha_1;\vec{\gamma}]+M[\alpha_2;\vec{\gamma}]\\&\quad-\{M[\alpha_1;\alpha_2]-M[\alpha_1;\alpha_2/\vec{\gamma}]\}\ .\end{aligned} \tag{9}$$

The last two terms in curly brackets can be interpreted as the mutual information between α_1 and α_2 that results from the input $\vec{\gamma}$ (unconstrained mutual information $M[\alpha_1;\alpha_2]$ minus the mutual information $M[\alpha_1;\alpha_2/\vec{\gamma}]$, which is not generated by $\vec{\gamma}$, hence for fixed $\vec{\gamma}$). The last term $M[\alpha_1;\alpha_2/\vec{\gamma}]$ disappears in a mean field description or any deterministic working network without additional input except $\vec{\gamma}$ (Sec. IV). Deterministic means that all entropies are zero if $\vec{\gamma}$ is fixed. The remaining three terms can be interpreted as above: Both neurons should transmit maximum but mutually independent information. The first two terms can be interpreted as enforcing cooperation between neurons of successive layers, the third leads to competition among neurons within one layer.

From $-M[\alpha_1;\alpha_2]$ an anti-Hebbian learning rule for the recurrent weights can be derived (details in Sec. V). Many models [12,5,4,3] assume anti-Hebbian learning within one layer. The minus sign in front of $M[\alpha_1;\alpha_2]$ in Eq. (9) can be seen as the reason for this.

We have split the mutual information for two output neurons into different terms, which have an easy interpretation. The main features of the learning rule originating from these terms are Hebbian learning for the feedforward weights and anti-Hebbian learning for the recurrent weights. In the following we will discuss in a more detailed way to what extent the strategy sketched here can be generalized to any number of output neurons. The aim is to establish local learning rules. Possibilities and limitations of local learning rules will become evident.

IV. MUTUAL INFORMATION, REDUNDANCY, AND LIMITATIONS OF LOCAL LEARNING RULES

First we extend Eq. (9) to the case of many output neurons. In the same way, by expanding the fraction in the logarithm we get

$$\begin{aligned} M[\vec{\alpha};\vec{\gamma}] &= \left\langle \ln \left\{ \frac{P_{\vec{\alpha}/\vec{\gamma}}}{P_{\vec{\alpha}}} \frac{\prod_i (P_{\alpha_i \vec{\gamma}} P_{\alpha_i})}{\prod_i (P_{\vec{\gamma}} P_{\alpha_i/\vec{\gamma}} P_{\alpha_i})} \right\} \right\rangle \\ &= \sum_i \left\langle \ln \frac{P_{\alpha_i \vec{\gamma}}}{P_{\alpha_i} P_{\vec{\gamma}}} \right\rangle - \left\langle \ln \frac{P_{\vec{\alpha}}}{\prod_i P_{\alpha_i}} \right\rangle \\ &\quad + \left\langle \ln \frac{P_{\vec{\alpha}/\vec{\gamma}}}{\prod_i P_{\alpha_i/\vec{\gamma}}} \right\rangle \\ &= \sum_i M[\alpha_i;\vec{\gamma}] - R[\vec{\alpha}] + R[\vec{\alpha}/\vec{\gamma}] \ . \end{aligned} \tag{10}$$

The term

$$R[\vec{\alpha}] = \left\langle \ln \frac{P_{\vec{\alpha}}}{\prod_i P_{\alpha_i}} \right\rangle \tag{11}$$

is often called the redundancy of the probability distribution $P_{\vec{\alpha}}$ and $R[\vec{\alpha}/\vec{\gamma}]$ is the redundancy of the Boltzmann distribution $P_{\vec{\alpha}/\vec{\gamma}}$ averaged over $\vec{\gamma}$. The difference $R[\vec{\alpha}] - R[\vec{\alpha}/\vec{\gamma}]$ may be interpreted as the redundancy at the output $\vec{\alpha}$ that is generated by the input $\vec{\gamma}$; i.e., the total redundancy minus that corresponding to thermal noise.

The minus sign in front of $R[\vec{\alpha}]$ in Eq. (10) requires a low statistical dependence of the output neurons for high mutual information. Redundancy reduction at the output is inherent in maximizing the mutual information between input and output. Some authors base their theory mainly on redundancy reduction [9]. However, somewhat different definitions for redundancy are used and additional constraints for the available information at the output have to be introduced. In this way an optimization principle for linear neurons is stated in [13,14] to describe properties of the information processing in the visual pathway.

For a given external input $\vec{\gamma}$ our network will work similarly to a mean field Boltzmann machine [15,16]. We will calculate the mean value $\langle \alpha_i \rangle^{\vec{\gamma}}$ of a neuron α_i for a given $\vec{\gamma}$ as a solution of

$$\begin{aligned} \langle \alpha_i \rangle^{\vec{\gamma}} &= \langle \tanh(h_{\alpha_i}) \rangle^{\vec{\gamma}} \\ &\approx \tanh(\langle h_{\alpha_i} \rangle^{\vec{\gamma}}) \quad \text{(mean field approximation)} \\ &= \tanh(\vec{w}_i \cdot \langle \vec{\alpha} \rangle^{\vec{\gamma}} + \vec{f}_i \cdot \vec{\gamma}) \ . \end{aligned} \tag{12}$$

The last equality suggests that with the dominant feedforward part $F\vec{\gamma}$ of the fields $\vec{h} = W\vec{\alpha} + F\vec{\gamma}$, the quality of the mean field (MF) approximation is increasing.

We use the mean field approximation to approximate the learning rule as well. Within a mean field theory only the self-consistently calculated mean values of Eq. (12) are used to describe the system. As in Eq. (12) we treat any expression $\langle g(\vec{\alpha}) \rangle^{\vec{\gamma}}$ not as an average but as a function of the mean values $g(\langle \vec{\alpha} \rangle^{\vec{\gamma}})$ and for any multipoint correlation function we have

$$\left\langle \prod_{i \in S} \alpha_i \right\rangle^{\vec{\gamma}} = \prod_{i \in S} \langle \alpha_i \rangle^{\vec{\gamma}}, \quad S \subset \{1, \ldots, N\} \ . \tag{13}$$

With the arguments of the Appendix this is equivalent to the statistical independence of the output neurons $\vec{\alpha}$ for a given input $\vec{\gamma}$

$$P_{\vec{\alpha}/\vec{\gamma}} = \prod_i P_{\alpha_i/\vec{\gamma}} \ . \tag{14}$$

This is the most general form of the mean field theory [17] and can be viewed as a maximum entropy assumption within an approximated description of the system by mean values only. Within this mean field approximation the redundancy of the Boltzmann distribution $R[\vec{\alpha}/\vec{\gamma}]$ vanishes. The same manipulations which lead to the different terms of Eq. (10) can be applied to Eq. (6) to split the learning rule into the contributions of the different terms discussed here. Therefore the contribution of $R[\vec{\alpha}/\vec{\gamma}]$ to the learning rule vanishes as well.

$R[\vec{\alpha}/\vec{\gamma}]$ describes the deviation from a mean field description. It reflects collective thermal fluctuations around mean values. $R[\vec{\alpha}/\vec{\gamma}]$ is zero if there are no thermal fluctuations or if the spins fluctuate statistically independently around their mean values. We assume that our MF approximation is increasingly valid with an increasing number of neurons, increasing external fields, and low enough temperature. Decreasing temperature corresponds to increasing weights, which is the general tendency during learning. The network is getting rid of its internal thermal noise by increasing weights.

Omitting $R[\vec{\alpha}/\vec{\gamma}]$, we can write for the mutual information (4)

$$\begin{aligned} M[\vec{\alpha};\vec{\gamma}] &\approx \sum_i M[\alpha_i;\vec{\gamma}] - R[\vec{\alpha}] \\ &\qquad \text{(mean field approximation)} \\ &= \sum_i \langle \ln(P_{\alpha_i/\vec{\gamma}}) \rangle - \langle \ln(P_{\vec{\alpha}}) \rangle \ . \end{aligned} \tag{15}$$

Every single neuron has to transmit high information about the input; however, simultaneously the global output distribution $P_{\vec{\alpha}}$ should have low redundancy and corresponding high entropy. While the first terms $\sum_i M[\alpha_i;\vec{\gamma}]$ concern single output neurons and therefore require only local information (Sec. V), the other term being a function of $P_{\vec{\alpha}}$ demands nonlocal information. This is most obvious if we express $P_{\vec{\alpha}}$ by its multipoint correlation functions $\langle \alpha_i \alpha_j \alpha_k \cdots \rangle$ also termed as moments (see the Appendix for the proof):

$$\begin{aligned}P_{\vec{\alpha}}=\frac{1}{2^N}\Bigg[&1+\sum_i \alpha_i\langle\alpha_i\rangle+\sum_{(ij)}\alpha_i\alpha_j\langle\alpha_i\alpha_j\rangle\\ &+\sum_{(ijk)}\alpha_i\alpha_j\alpha_k\langle\alpha_i\alpha_j\alpha_k\rangle+\cdots\Bigg]\\ =2^{-N}&\sum_{S\subset\{1,2,\ldots,N\}}\prod_{i\in S}\alpha_i\left\langle\prod_{i\in S}\alpha_i\right\rangle . \qquad (16)\end{aligned}$$

$\sum_{(ij)}$ and $\sum_{(ijk)}$ respectively denote the sum over all possible pairs and triplets, respectively, of pairwise different indices or generally $\sum_{S\subset\{1,2,\ldots,N\}}$ is the sum over all possible subsets $S\subset\{1,2,\ldots,N\}$. The entropy $\langle -\ln(P_{\vec{\alpha}})\rangle$ reaches its maximum if all correlation functions vanish. To maximize $\langle -\ln(P_{\vec{\alpha}})\rangle$ we need information about all multipoint correlations. But correlation functions with more than two neurons have to be interpreted as nonlocal quantities.

In general it will not be possible to determine all multipoint correlations independently in a network of N spins with the topology of Fig. 1. By using a bias we can control mean values. The $N(N-1)/2$ recurrent weights W can be used to determine the $N(N-1)/2$ two-point correlations. In order to additionally control higher multipoint correlations independently we would have to introduce corresponding multipoint interactions. With the two-point interaction of the energy function (1) only and the topology of Fig. 1, whether appropriate weights exist in a way such that the output approximately reaches the maximum entropy of $N\ln(2)$ depends on the input distribution $P_{\vec{\gamma}}$. Hence, after learning even with the exact learning rule (6) the amount of information extracted from the input distribution is not determined *a priori.* In general some redundancy will remain at the output layer.

For some tasks vanishing redundancy is even not desirable, e.g., we could be interested in storing the states of the output layer in an associative network. Gardner's theory of connections (see, e.g., [18]) provides the limits for the number of storable states dependent on the number of units N used. A lot of redundancy among the set of states to be stored is necessary for the memory to work. Furthermore, a redundancy free representation is not robust against damage of single neurons.

Altogether, instead of trying to reach vanishing higher-order correlations corresponding to vanishing redundancy by nonlocal learning rules, which is not possible in general with two-point interactions only, we think it advisable to restrict our study to local learning rules and remove the first two moments only. This might cause some additional redundancy and we have to raise the number of output neurons correspondingly to reach some desired entropy $\langle -\ln(P_{\vec{\alpha}})\rangle$ at the output. It should be possible to prove that a probability distribution (16) on $\{-1,+1\}^N$ with vanishing mean values $\langle\alpha_i\rangle$ and vanishing two-point correlations $\langle\alpha_i\alpha_j\rangle$ has a minimum entropy depending on N that can be raised to any arbitrary amount by raising N. Additionally, we will see that it is possible to remove all mean values and two-point correlations with the aid of a bias and the recurrent weights.

For two spins α_i and α_j statistical independence $P_{\alpha_i\alpha_j}=P_{\alpha_i}P_{\alpha_j}$ is equivalent to vanishing covariance $\langle\alpha_i\alpha_j\rangle-\langle\alpha_i\rangle\langle\alpha_j\rangle=0$ (see the Appendix). Hence, removing the covariance for all pairs (ij) results in pairwise, statistically independent neurons. The corresponding cost function is the sum over the mutual information between all pairs of neurons $\sum_{(ij)}M[\alpha_i;\alpha_j]$. Pairwise statistical independence is a less strong requirement not equivalent to the total independence of all neurons $(P_{\vec{\alpha}}=\prod_i P_{\alpha_i})$ with $R[\vec{\alpha}]$ the corresponding cost function. Using only local information to remove the covariance $\langle\alpha_i\alpha_j\rangle-\langle\alpha_i\rangle\langle\alpha_j\rangle$ is equivalent to replacing $R[\vec{\alpha}]$ in (15) by $\sum_{(ij)}M[\alpha_i;\alpha_j]$:

$$K\equiv\sum_i M[\alpha_i;\vec{\gamma}]-\lambda\sum_{(ij)}M[\alpha_i;\alpha_j] . \qquad (17)$$

For two output neurons this cost function is the mutual information (9) if we neglect $M[\alpha_1;\alpha_2/\vec{\gamma}]$ within the mean field approximation. The factor λ can be used to keep the balance between cooperation and competition in cases where the maximum of the two parts of this cost function cannot be reached simultaneously. We will see in Sec. V that a learning rule that is essentially Hebbian learning for the feedforward weights and anti-Hebbian learning for the recurrent weights can be deduced as the gradient in this cost function with minor approximations.

For simplicity we will restrict our study to vanishing mean values in the following. For example, for a symmetric input distribution $P(\vec{\gamma})=P(-\vec{\gamma})$ all odd moments vanish, for everything is symmetric under a change of sign of all units. In the general case any of the following expressions can be extended to nonvanishing mean values with some additional analysis. Numerical experiments get somewhat more expensive, for mean values have to be calculated and stored [see, for example, Eq. (8)]. Additionally, for nonsymmetric $P_{\vec{\gamma}}$, a bias $\beta=1$ should be added to get rid of the mean values with the anti-Hebbian learning rule for the weight b_i:

$$\Delta b_i=-\eta\langle\beta\alpha_i\rangle=-\eta\langle\alpha_i\rangle . \qquad (18)$$

With vanishing mean value every single neuron has a high entropy, which is enforced by the first sum in (17) and (15). For any $P_{\vec{\gamma}}$, F, and W there exists a bias $\vec{b}$ that makes the mean values and the learning rule (18) will find this solution.

V. DERIVATION OF A LEARNING RULE BY THE COST FUNCTION (17)

We will now calculate the corrections resulting from the cost function (17) for the weights $\vec{f}_1$ and w_{12} (see Fig. 1) for the eight representative contributions: $M[\alpha_i;\vec{\gamma}]$, $i=1,2,3,4$; $M[\alpha_1;\alpha_2]$; $M[\alpha_2;\alpha_3]$; $M[\alpha_3;\alpha_4]$; and $M[\alpha_4;\alpha_1]$. These eight terms contain all possible cases, e.g., how the weight w_{12} is affected by the mutual information $M[\alpha_3;\alpha_4]$ between the two other neurons α_3

TABLE I. Contributions to the correction of $\vec{f}_1$ and w_{12} for vanishing mean values from different representative mutual informations. Under the results the letters MF indicate that mean field equations have been used.

To / From	$\vec{\nabla}_{f_1}$	$\frac{\partial}{\partial w_{12}}$	
	$M[\alpha_1;\vec{\gamma}]$	$\left\langle \frac{\vec{\gamma}\langle h_{\alpha_1}\rangle^{\vec{\gamma}}}{(\cosh\langle h_{\alpha_1}\rangle^{\vec{\gamma}})^2} \right\rangle = o(\mathcal{C}^0)$ MF	$\left\langle \frac{\langle\alpha_2\rangle^{\vec{\gamma}}\langle h_{\alpha_1}\rangle^{\vec{\gamma}}}{(\cosh\langle h_{\alpha_1}\rangle^{\vec{\gamma}})^2} \right\rangle = o(\mathcal{C}^1)$ MF
	$M[\alpha_2;\vec{\gamma}]$	0 MF	$\left\langle \frac{\langle\alpha_2\rangle^{\vec{\gamma}}\langle h_{\alpha_2}\rangle^{\vec{\gamma}}}{(\cosh\langle h_{\alpha_2}\rangle^{\vec{\gamma}})^2} \right\rangle = o(\mathcal{C}^1)$ MF
	$M[\alpha_3;\vec{\gamma}]$	0 MF	0 MF
	$M[\alpha_4;\vec{\gamma}]$	0 MF	0 MF
	$M[\alpha_1;\alpha_2]$	$\langle\vec{\gamma}[1-(\langle\alpha_1\rangle^{\vec{\gamma}})^2]\rangle \operatorname{arctanh}\langle\alpha_1\alpha_2\rangle = o(\mathcal{C}^1)$	$\langle 1-(\langle\alpha_1\alpha_2\rangle^{\vec{\gamma}})^2\rangle \operatorname{arctanh}\langle\alpha_1\alpha_2\rangle = o(\mathcal{C}^1)$
	$M[\alpha_1;\alpha_3]$	$\langle\vec{\gamma}[1-(\langle\alpha_1\rangle^{\vec{\gamma}})^2]\rangle \operatorname{arctanh}\langle\alpha_1\alpha_3\rangle = o(\mathcal{C}^1)$	$\le o(\mathcal{C}^2)$
	$M[\alpha_2;\alpha_4]$	0 MF	$\le o(\mathcal{C}^2)$
	$M[\alpha_3;\alpha_4]$	0 MF	0 MF

and α_4, which results in a nonlocal contribution. The results are displayed in Table I.

We do the algebra explicit for the term $\vec{\nabla}_{f_1} M[\alpha_1;\vec{\gamma}]$. Analogously to the derivation of Eq. (6) we obtain

$$\vec{\nabla}_{f_1} M[\alpha_1;\vec{\gamma}] = \langle (\ln P_{\alpha_1/\vec{\gamma}} - \ln P_{\alpha_1})(\alpha_1\vec{\gamma} - \langle \alpha_1\vec{\gamma}\rangle^{\vec{\gamma}})\rangle \ . \quad (19)$$

First we evaluate the $\ln P_{\alpha_1/\vec{\gamma}}$ term using for the probability distribution the general expressions (16)

$$\begin{aligned}
\langle (\ln P_{\alpha_1/\vec{\gamma}})(\alpha_1\vec{\gamma} - \langle\alpha_1\vec{\gamma}\rangle^{\vec{\gamma}})\rangle &= \left\langle \sum_{\alpha_1} P_{\alpha_1/\vec{\gamma}} \ln(P_{\alpha_1/\vec{\gamma}})(\alpha_1\vec{\gamma} - \langle\alpha_1\rangle^{\vec{\gamma}}\vec{\gamma}) \right\rangle \\
&= \left\langle \vec{\gamma} \sum_{\alpha_1} (1+\alpha_1\langle\alpha_1\rangle^{\vec{\gamma}})(\alpha_1 - \langle\alpha_1\rangle^{\vec{\gamma}}) \frac{1}{2} \ln \frac{1+\alpha_1\langle\alpha_1\rangle^{\vec{\gamma}}}{2} \right\rangle \\
&= \left\langle \vec{\gamma}[1-(\langle\alpha_2\rangle^{\vec{\gamma}})^2] \frac{1}{2} \ln \frac{1+\langle\alpha_1\rangle^{\vec{\gamma}}}{1-\langle\alpha_1\rangle^{\vec{\gamma}}} \right\rangle \\
&= \langle \vec{\gamma}[1-(\langle\alpha_1\rangle^{\vec{\gamma}})^2] \operatorname{arctanh}\langle\alpha_1\rangle^{\vec{\gamma}}\rangle \\
&\approx \left\langle \vec{\gamma} \frac{\langle h_{\alpha_1}\rangle^{\vec{\gamma}}}{(\cosh\langle h_{\alpha_1}\rangle^{\vec{\gamma}})^2} \right\rangle \quad \text{(mean field approximation)} \ . \quad (20)
\end{aligned}$$

Only in the last equality has the mean field equation (12) been used. Up to this point the result is exact for a single neuron α_1. The $\ln(P_{\alpha_1})$ term in (19) can be treated in the same way. Together we get

$$\vec{\nabla}_{f_1} M[\alpha_1;\vec{\gamma}] = \left\langle \vec{\gamma} \frac{\langle h_{\alpha_1}\rangle^{\vec{\gamma}}}{(\cosh\langle h_{\alpha_1}\rangle^{\vec{\gamma}})^2} \right\rangle - \left\langle \frac{\vec{\gamma}}{(\cosh\langle h_{\alpha_1}\rangle^{\vec{\gamma}})^2} \right\rangle \operatorname{arctanh}\langle\alpha_1\rangle \ . \quad (21)$$

With vanishing mean values $\langle\vec{\alpha}\rangle = 0$ the term proportional to $\operatorname{arctanh}\langle\alpha_1\rangle$, which originates from the $\ln(P_{\alpha_1})$ term in (19), disappears. The results in Table I are obtained for vanishing mean values. The remaining contribution (20) is local and Hebb-like: it is the presynaptic activity $\vec{\gamma}$ times the postsynaptic field $\langle h_{\alpha_1}\rangle^{\vec{\gamma}}$ times a positive function of the postsynaptic field, which is $1/(\cosh\langle h_{\alpha_1}\rangle^{\vec{\gamma}})^2$. We interpret this function in the following way.

(i) For small postsynaptic potentials the neuron α_1 is not clearly dedicated to one of its two possible states. Therefore learning is necessary. This learning is Hebbian learning.

(ii) For large values of the postsynaptic potential $\langle h_{\alpha_1}\rangle^{\vec{\gamma}}$ there is a definite assignment of the pattern $\vec{\gamma}$ to one of the two classes, therefore learning is suppressed by the factor $1/(\cosh\langle h_{\alpha_1}\rangle^{\vec{\gamma}})^2$. This is similar to the perceptron learning where only input patterns with small stability parameter are used to adapt the weights.

Table I shows the other contributions to the correction of the weight $\vec{f}_1$ that can be calculated in a similar way in terms of the correlation functions by using the general expression (16) for any probability distribution. Most of them are zero if we use the mean field approximation $\langle \alpha_i \alpha_j \rangle^{\vec{\gamma}} = \langle \alpha_i \rangle^{\vec{\gamma}} \langle \alpha_j \rangle^{\vec{\gamma}}$. The two remaining terms proportional to arctanh $(\langle \alpha_i \alpha_j \rangle)$ vanish for a symmetric input distribution $P(\vec{\gamma}) = P(-\vec{\gamma})$. In the general case we may assume that the correlations $\langle \alpha_i \alpha_j \rangle$ are small, for our cost function forces the output neurons to be pairwise statistically independent. Expressions linear in $\langle \alpha_i \alpha_j \rangle^k$ are said to be of order k in statistical correlations (SCs): $\langle \alpha_i \alpha_j \rangle^k = o(\mathcal{C}^k)$. We will keep only the highest order for the adaption of every weight. This is plausible for the following reasons.

(i) We start learning with small weights, which corresponds to a high temperature. Therefore correlations are small at the beginning of learning. Additionally we could learn for some first epochs only with the decorrelating terms.

(ii) At the end of a successful learning, $M[\alpha_i;\alpha_j]$ will be small, which means low correlations $\langle \alpha_i \alpha_j \rangle$ of the output units.

(iii) To ensure low statistical correlations in the course of learning, we may use a large λ in the cost function (17).

The contribution (20) is independent of correlations of the output neurons and is therefore of the order $o(\mathcal{C}^0)$. Table I shows that all other contributions to the correction of $\vec{f}_1$ are $\leq o(\mathcal{C}^1)$. For the feedforward weights we will keep the $o(\mathcal{C}^0)$ term only.

The contributions for the correction of the recurrent weight w_{12} can be evaluated with similar techniques making extensive use of the relations of the Appendix. For the contribution from $M[\alpha_1;\alpha_2]$ the exact result is

$$\begin{aligned}\frac{\partial}{\partial w_{12}} M[\alpha_1;\alpha_2] &= \langle 1 - (\langle \alpha_1 \alpha_2 \rangle^{\vec{\gamma}})^2 \rangle \operatorname{arctanh}(\langle \alpha_1 \alpha_2 \rangle) \\ &= \langle 1 - (\langle \alpha_1 \alpha_2 \rangle^{\vec{\gamma}})^2 \rangle \langle \alpha_1 \alpha_2 \rangle + o(\mathcal{C}^2) \ .\end{aligned} \tag{22}$$

The factor $\Theta_{12} \equiv \langle 1 - (\langle \alpha_1 \alpha_2 \rangle^{\vec{\gamma}})^2 \rangle \in [0,1]$ is a positive number and depends on a local term only. Omitting this factor is essentially an enlargement of the parameter λ in (17), but results in the simple anti-Hebbian learning rule $\Delta w_{ij} \sim -\langle \alpha_i \alpha_j \rangle$. The factor λ is irrelevant if the maximum of the two parts of the cost function (17) can be reached simultaneously. Some of the experiments of Sec. VI were done with and without the factor Θ_{ij}. In any case we found equivalent results.

The learning rule corresponding to (22) as well as the pure anti-Hebbian learning $\Delta w_{ij} \sim -\langle \alpha_i \alpha_j \rangle$ without Θ_{ij} vanishes only if all correlations vanish: $\langle \alpha_i \alpha_j \rangle = 0$ for all pairs (ij). This is equivalent to $\sum_{(ij)} M[\alpha_i;\alpha_j] = 0$ (see the Appendix). Starting at any point where the SC approximation is valid, the anti-Hebbian learning with and without Θ_{ij} has a positive projection onto the exact gradient in $\sum_{(ij)} M[\alpha_i;\alpha_j]$. Hence this anti-Hebbian learning converges to a solution with $\sum_{(ij)} M[\alpha_i;\alpha_j] = 0$ without getting stuck in a local minimum where $\sum_{(ij)} M[\alpha_i;\alpha_j] \neq 0$. Near $\sum_{(ij)} M[\alpha_i;\alpha_j] = 0$ the SC approximation is almost exact.

By inspecting the results in Table I it becomes clear that there are two other $o(\mathcal{C}^1)$ contributions to the correction of the weight w_{12} that result from $M[\alpha_1;\vec{\gamma}]$ and $M[\alpha_2;\vec{\gamma}]$. These terms are local too, but lead to correlations between α_1 and α_2 and compete with the term (22) resulting from $M[\alpha_1;\alpha_2]$. Indeed for $\lambda < 2$ and high temperature (small weights) $(\partial/\partial w_{12})/(M[\alpha_1;\vec{\gamma}] + M[\alpha_2;\vec{\gamma}])$ exceeds $-(\partial/\partial w_{12}) M[\alpha_1;\alpha_2]$. This is due to the fact that for high temperature (high noise) it is better to transmit the most important feature of the input twice instead of an additional less important feature. For the low-temperature case at the end of learning the term $-(\partial/\partial w_{12}) M[\alpha_1;\alpha_2]$ is dominant. Hence, to avoid high correlations in the course of learning we always used $\lambda > 2$.

VI. EXPERIMENTS

Summing up our learning rule consists of a Hebbian term for any weight that is suppressed by the factor $1/(\cosh\langle h_\alpha \rangle^{\vec{\gamma}})^2$ for large postsynaptic fields $\langle h_\alpha \rangle^{\vec{\gamma}}$. This factor is not a global factor but specific for every $\vec{\gamma}$. Additionally the weights within one layer are adapted by the anti-Hebbian term $\Delta w_{ij} \sim -\Theta_{ij} \langle \alpha_i \alpha_j \rangle$. Θ_{ij} is a positive number independent of $\vec{\gamma}$ and therefore may be omitted. If necessary we add a bias to keep the mean values small.

Learning does not converge (only for $T=0$, which corresponds to infinite weights), but becomes infinitely slow for the factors $1/(\cosh\langle h_\alpha \rangle^{\vec{\gamma}})^2$ and Θ_{ij}. Convergence could be achieved by using an arbitrarily small decay term for the weights. We terminated learning if there was no considerable progress anymore.

At the end of any experiment we calculated

$$\begin{aligned} C &\equiv \langle \vec{\alpha} \vec{\alpha}^T \rangle \\ &\approx \langle \langle \vec{\alpha} \rangle^{\vec{\gamma}} \langle \vec{\alpha}^T \rangle^{\vec{\gamma}} \rangle \quad \text{(mean field approximation)} \end{aligned}$$

and $\langle \vec{\alpha} \rangle$ to control the success of learning. In any case we found $C_{ij} < 0.01$ for the nondiagonal elements and $C_{ii} > 0.95$ for the diagonal elements of C and $\langle \alpha_i \rangle < 0.003$, except for the "retina model" (see below), where we found $C_{ii} \approx 0.9$ and $C_{ij} < 0.03$.

Additionally, when calculating the mean values of Eq. (12) we should use a mean field annealing schedule as proposed in [15]. Due to the dominating feedforward part $F\vec{\gamma}$ of the field $\langle \vec{h} \rangle^{\vec{\gamma}} = W \langle \vec{\alpha} \rangle^{\vec{\gamma}} + F\vec{\gamma}$ we found no influence of an annealing schedule and calculated the mean fields $\langle \vec{\alpha} \rangle^{\vec{\gamma}}$ by iterating Eq. (12) in parallel until some convergence criterion for fixed temperature $T=1$. Initial values had no influence on the solution of (12) for in our examples the system is mainly driven by the external field $\vec{\gamma}$. This may not be the general case. In addition, the parallel iteration of Eq. (12), which corresponds to integrating

$$\frac{d}{dt} \langle \alpha_i \rangle^{\vec{\gamma}}(t) = -\langle \alpha_i \rangle^{\vec{\gamma}}(t) + \tanh(\vec{w}_i \cdot \langle \vec{\alpha} \rangle^{\vec{\gamma}}(t) + \vec{f}_i \cdot \vec{\gamma}) \tag{23}$$

with step size $\Delta t = 1$, may not converge. But for small enough step size the system (23) converges always to a stationary state $(d/dt)\langle \alpha_i \rangle^{\vec{\gamma}}(t) = 0$ for there exists a Liapunov function that always decreases under the dynamics of (23) [19]. For a more general stability analysis see [20]. A stationary state of (23) is a solution of (12). In our case this Liapunov function is the "mean field free energy"

$$F_{\vec{\gamma}}^{\mathrm{MF}} = -\langle \vec{\alpha}^T \rangle^{\vec{\gamma}} \cdot F \vec{\gamma} - \tfrac{1}{2} \langle \vec{\alpha}^T \rangle^{\vec{\gamma}} \cdot W \langle \vec{\alpha} \rangle^{\vec{\gamma}} + \sum_i \left\{ \frac{1+\langle \alpha_i \rangle^{\vec{\gamma}}}{2} \ln \frac{1+\langle \alpha_i \rangle^{\vec{\gamma}}}{2} + \frac{1-\langle \alpha_i \rangle^{\vec{\gamma}}}{2} \ln \frac{1-\langle \alpha_i \rangle^{\vec{\gamma}}}{2} \right\}, \tag{24}$$

and the dynamics (23) is with positive projection along the gradient of $F_{\vec{\gamma}}^{\mathrm{MF}}$. This expresses the consistency of the mean field theory: the mean values $\langle \vec{\alpha} \rangle^{\vec{\gamma}}$ are calculated in such a way that the corresponding mean field distribution $\prod_i \frac{1}{2}(1+\alpha_i \langle \alpha_i \rangle^{\vec{\gamma}})$ minimizes the mean field free energy $F_{\vec{\gamma}}^{\mathrm{MF}}$ among all distributions of the form $P_{\vec{\alpha}} = \prod_i P_{\alpha_i}$.

A. One output neuron

For a single neuron the expression (21) is exact. Figure 2 shows the evolution of the weight vector in a two-dimensional input space for different initial values. As input we chose (1,1) and $(-1,-1)$ with a probability 0.4 and $(1,-1)$ and $(-1,1)$ with a probability 0.1. $P_{\vec{\gamma}}$ is symmetric and the mean value of the output neuron vanishes. For small weights (large temperature) the neuron "sees" only a hazy input distribution and heads for the principal component along the diagonal (1,1). For low temperature this is not optimal because with the weight vector along (1,1), the inputs $(1,-1)$ and $(-1,1)$ are not assigned definitely to one of the two classes. Hence the weight vector begins to deviate from the diagonal. Figure 3 shows the mutual information dependent on the norm and the orientation of the weight vector in the two-dimensional input plane. While for small weights the first principal component of the input distribution at 45° in Fig. 3 is the direction of maximum mutual information, this direction corresponds to a local minimum for large weights. For one linear neuron similar examples can be constructed where the first principal component is different from the direction of maximal mutual information.

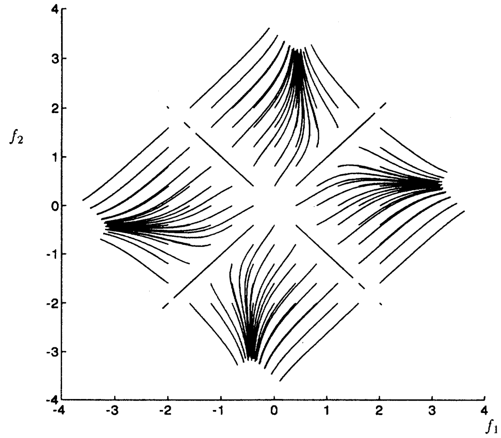

FIG. 2. Phase flow of the gradient field of the mutual information in the two-dimensional weight space $(f_1, f_2) \in \mathbb{R}^2$ for a single output neuron. The input distribution on $\{\pm 1\}^2$ is described in the text. The direction of the flow is always oriented towards increasing weights.

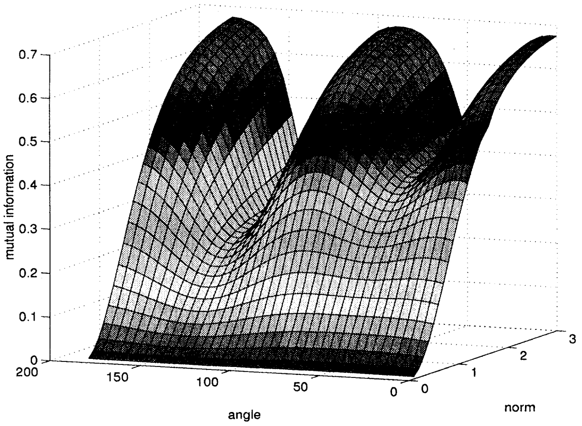

FIG. 3. Mutual information (bits) for a single output neuron plotted over the angle to the x axis (deg) and the norm of the weight $\vec{f}$. The input distribution on $\{\pm 1\}^2$ is described in the text.

B. Two output neurons

We have drawn samples from a probability distribution in two dimensions consisting of four equal Gaussians located at (1,1), $(1,-1)$, $(-1,1)$, and $(-1,-1)$. Figure 4 shows the samples used and the way a network with two output neurons classifies them into four classes.

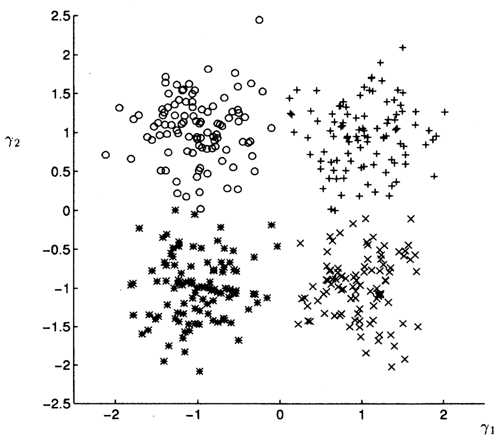

FIG. 4. Classification of 400 two-dimensional inputs (γ_1, γ_2) drawn from a probability distribution consisting of four equal Gaussians. The four different symbols mark the four different responses of the two binary output units.

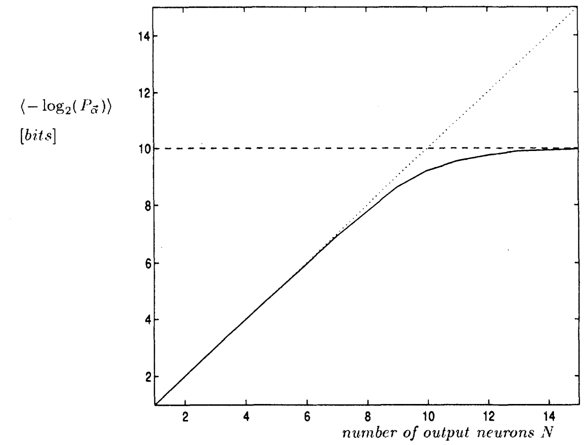

FIG. 5. Full line, the reached entropy at the output $\langle -\log_2(P_{\vec{\alpha}})\rangle$, dependent on the number of output spins N; horizontal dashed line, the entropy of the input $\log_2(2^{10})=10$; dotted line with slope 1, maximum possible entropy $\log_2(2^N)=N$ of N spins.

C. 2^{10} random patterns

As input we chose 2^{10} random 30-dimensional patterns $\vec{\gamma}\in[-1,1]^{30}$ and presented them with equal probability to a network with N output spins. We used $N=1,2,\ldots,15$ and calculated the entropy $\langle -\log_2(P_{\vec{\alpha}})\rangle$ at the end of learning, which is nearly the mutual information (4), for $\langle \log_2(P_{\vec{\alpha}/\vec{\gamma}})\rangle$ is vanishing in the limit of vanishing temperature. Figure 5 shows the result averaged over three trials every time with new random input pattern. Also shown are the entropy of the input distribution [horizontal line at $\log_2(2^{10})=10$] and the maximum entropy of N spins [the line $\log_2(2^N)=N$]. For a small number of output neurons $\langle -\log_2(P_{\vec{\alpha}})\rangle$ is identical to this maximum. There are only small combinatorical possibilities to form multipoint correlations that could generate redundancy at the output. With increasing N the entropy $\langle -\log_2(P_{\vec{\alpha}})\rangle$ approaches the horizontal line $\log_2(2^{10})=10$. At $N=15$ we have reached nearly maximal mutual information and a nearly bijective mapping of every input $\vec{\gamma}$ onto an internal representation $\vec{\alpha}$.

D. A very simple model for the retina

The first step of information processing of visual data happens in the retina itself. Before the ganglion cells send their signals along the optic nerve, a part of the redundancy of the visual information incident on the rods and cones is removed by the system of horizontal, bipolar, and amacrine cells. This redundancy is to a large part caused by the correlation of nearby image pixels when looking at typical scenes in our environment. Mainly the horizontal cells serve for lateral interaction and thereby enhance visual contrast. Every ganglion cell still gives a local response; hence we can attribute to every ganglion cell a local receptive field.

We used the following very simple model for the retina. The input is a string of 100 bits where every bit is correlated with its next neighbor. This is achieved by drawing 100 random numbers in every learning epoch, adding every number to its next neighbor, and taking the sign of the result. Every output neuron (which should correspond to a ganglion cell in this rough simplification) is connected to all other output neurons and to ten neighboring input neurons only (rods and cones). These receptive fields of our ganglion cells overlap by five input bits. The input distribution is symmetric; hence without a bias the odd moments vanish. Additionally multipoint correlations of order ≥ 4 vanish, for in any set of four neurons the receptive fields of at least two neurons are separated by more than one correlation length of the input. Hence, within the special fixed topology of this model and for the described input distribution our learning rule should result in the maximum possible mutual information corresponding to vanishing redundancy at the output.

After learning we recorded the result by moving one "illuminated" pixel along the string of input bits. Figure 6 shows the mean response $\langle \alpha_i\rangle^{\vec{\gamma}}$ of three representative output neurons. These "sombrero" response functions are well known from the physiology of the eye and can be deduced using information-theoretic arguments (see [13] and the references therein).

Illuminating only one pixel is not a typical input but just a way to record the result. The network is trained to operate optimally on inputs that rather look like "blocks of equal image pixels." Figure 7 shows the response $\langle \alpha_i\rangle^{\vec{\gamma}}$ of one output neuron on moving 20 neighboring "illuminated" image pixels along a "dark" background. The displayed output neuron responds mainly on edges of blocks of equal inputs. All information is contained in the location of those edges. Recording only the location of the edges is one possible way of transforming the input signal into a representation with less redundancy. It is interesting to note that indeed the first one to observe effects corresponding to Fig. 7 was Mach in 1865 [21]. These observations led him to the conclusion that there

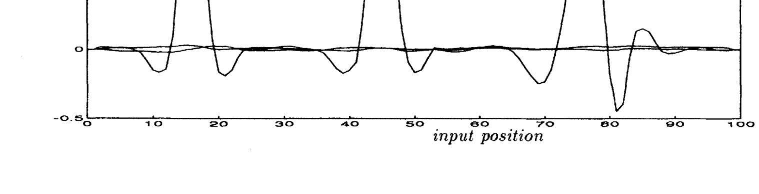

FIG. 6. Mean response $\langle \alpha_i\rangle^{\vec{\gamma}}$ of three "ganglion cells" on moving one illuminated pixel $\gamma_j=1$ along the input. All other image pixels are set to 0. The horizontal axis is the position of the illuminated pixel in the 100-dimensional input.

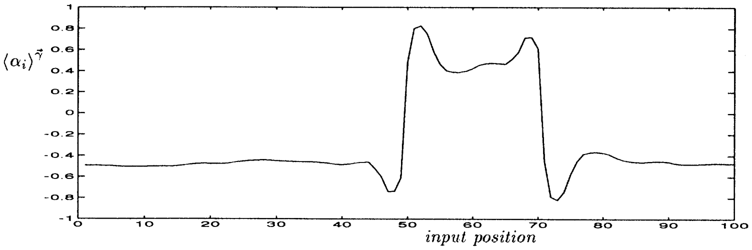

FIG. 7. Mean response $\langle\alpha_i\rangle^{\vec{\gamma}}$ of one ganglion cell on moving 20 on pixels ($\gamma_i=+a$) along the input. All other image pixels are set to off ($\gamma_i=-a$). We have scaled the input to $a=0.2$ to prevent the output units from saturating too fast *at* ±1.

must be a lateral interaction within the retina.

We would like to stress again that it is not our intention to exactly model biology but rather to show that with our simplified model based on a information-theoretic background it is possible to produce well known and plausible results.

VII. SUMMARY AND OUTLOOK

We have considered local learning rules from an information-theoretic point of view in a recurrent network of Ising spins. The mutual information is a function of the global probability distributions $P_{\vec{\alpha}}$, which depends on nonlocal expressions [Eq. (16)]. Therefore nonlocal terms (e.g., multipoint correlations) are hardly avoidable when constructing learning rules that maximize the mutual information. Additionally, a bias and two-point interactions are not sufficient to determine all multipoint correlations independently. The possible mappings our network can represent are restricted by using only two-point interactions. Hence, even if we use nonlocal expressions, which are expensive to calculate, we cannot be sure to reach vanishing redundancy and the maximum possible "information density" $\equiv\langle-\ln(P_{\vec{\alpha}})\rangle/N=\ln(2)$ with N spins at the output layer.

The local learning rules proposed here are not aimed at this maximum. Local learning rules corresponds to a cost function (17), which is not equivalent to the mutual information (4). This cost function forces the output neurons to be pairwise statistically independent instead of totally independent output neurons. The consideration of pairwise statistical dependences corresponds to the use of pairwise couplings only. The resulting anti-Hebbian learning rule uses these two-point interactions to determine corresponding two-point correlations. In contrast to the exact learning rule, which will not necessarily reach $R[\vec{\alpha}]=0$, the anti-Hebbian learning will always result in $\sum_{(ij)}M[\alpha_i;\alpha_j]=0$. In this way any amount of information can be extracted from the input if we use sufficient output neurons. But our simple learning rules are obtained at the cost of some additional redundancy in the way the information is coded at the output. However, from the viewpoint of biology or if we want to store the states of the output layer in an associative memory, a redundancy free coding might not even be desirable.

Within these limitations there are no further restrictions; especially no assumptions on the input distribution are made. The very simple local learning rules together with the deterministic mean field dynamics result in a very fast working network that can be used with a large number of neurons.

Before using this formalism on a large scale, the validity of the mean field approximation (13) and the corresponding dynamics (23) should be investigated in more detail. The mean field theory faces two independent problems. First, the mean field distribution (13) might not well approximate the Boltzmann distribution. Second, the dynamics (23) might not find the global minimum of the mean field free energy (24), but get stuck in a local minimum. However, our experimental results are encouraging, which is due to the strong feedforward part $F\vec{\gamma}$ of the fields $W\vec{\alpha}+F\vec{\gamma}$ in our network topology. Some investigations of deterministic mean field methods have already been done in [22–24].

Another interesting extension of our work might be to translate the formalism presented here to spins with states 0 and 1 (instead of ±1). The bias is then controlling the mean activity and can be used to switch continuously from coding with low activity (Pott spinlike states) to a distributed coding with equal probability for 0 and 1. For low activity the lateral interaction should mainly develop as lateral inhibition. From physiological observations lateral and temporal inhibition is obvious in sensory processing to enhance lateral and temporal contrast. Földiak intuitively went this way in [12]. 0-1 coding with low activity could have some advantages over ±1 coding with $\langle\vec{\alpha}\rangle=0$ and is biologically more appealing.

APPENDIX: USEFUL RELATIONS FOR PROBABILITY DISTRIBUTIONS ON $\{-1,+1\}^N$

The probability distribution P_α for one spin α with the binary values of ±1 is determined by its mean value alone

$$P_\alpha=\tfrac{1}{2}(1+\alpha\langle\alpha\rangle)\ . \tag{A1}$$

The generalization of this expression to N spins is

$$P_{\vec{\alpha}}=2^{-N}\sum_{S\subset\{1,2,\ldots,N\}}\prod_{i\in S}\alpha_i\left\langle\prod_{i\in S}\alpha_i\right\rangle\ . \tag{A2}$$

$\sum_{S\subset\{1,2,\ldots,N\}}$ is the sum over all possible subsets $S\subset\{1,2,\ldots,N\}$ including the empty set [see also Eq. (16)]. To prove this equation it is enough to state that any probability distribution on $\{-1,+1\}^N$ is unique, determined by its finite number of 2^N different multipoint correlations $\langle\cdots\alpha_i\alpha_j\cdots\rangle$, and the correlations produced by (A2) are the desired ones. In the sum $\sum_{\vec{\alpha}}P_{\vec{\alpha}}\cdots\alpha_i\alpha_j\cdots$ the only symmetric term that survives the summation is $\langle\cdots\alpha_i\alpha_j\cdots\rangle$. Equation (A2) can also be proved by induction. To see this we first state another useful relation.

An n-point correlation function for the spins $\alpha_1 \cdots \alpha_n$, given the state of one additional spin β

$$\langle \alpha_1 \cdots \alpha_n \rangle^\beta = \sum_\alpha P_{\alpha_1 \cdots \alpha_n/\beta} \alpha_1 \cdots \alpha_n \ ,$$

can be expressed as

$$\langle \alpha_1 \cdots \alpha_n \rangle^\beta = \frac{\langle \alpha_1 \cdots \alpha_n \rangle + \beta \langle \alpha_1 \cdots \alpha_n \beta \rangle}{1+\beta\langle \beta \rangle} = \frac{\langle \alpha_1 \cdots \alpha_n \rangle + \beta \langle \alpha_1 \cdots \alpha_n \beta \rangle}{2P_\beta} \ . \qquad \text{(A3)}$$

Proof. We have

$$\langle \alpha_1 \cdots \alpha_n \beta \rangle = \sum_\beta P_\beta \beta \langle \alpha_1 \cdots \alpha_n \rangle^\beta = \tfrac{1}{2}\sum_\beta (1+\beta\langle\beta\rangle)\beta\langle \alpha_1 \cdots \alpha_n \rangle^\beta$$

and

$$\langle \alpha_1 \cdots \alpha_n \rangle = \sum_\beta P_\beta \langle \alpha_1 \cdots \alpha_n \rangle^\beta = \tfrac{1}{2}\sum_\beta (1+\beta\langle\beta\rangle)\langle \alpha_1 \cdots \alpha_n \rangle^\beta \ .$$

From these two equations we get

$$\langle \alpha_1 \cdots \alpha_n \beta \rangle + \langle \alpha_1 \cdots \alpha_n \rangle = (1+\langle\beta\rangle)\tfrac{1}{2}\sum_\beta (\beta+1)\langle \alpha_1 \cdots \alpha_n \rangle^\beta = (1+\langle\beta\rangle)\langle \alpha_1 \cdots \alpha_n \rangle^{\beta=+1} \ ,$$

$$\langle \alpha_1 \cdots \alpha_n \beta \rangle - \langle \alpha_1 \cdots \alpha_n \rangle = (\langle\beta\rangle - 1)\langle \alpha_1 \cdots \alpha_n \rangle^{\beta=-1}.$$

The last two equations can be combined to the desired result.

Now it is easy to prove Eq. (A2) by induction. For $N=1$ we have Eq. (A1). The induction step is as follows: For $\vec{\alpha} \in \{-1,+1\}^N$ and an additional spin $\alpha_{N+1} \equiv \beta$ we have

$$\begin{aligned} P_{\vec{\alpha}\beta} = P_\beta P_{\vec{\alpha}/\beta} &= P_\beta 2^{-N} \sum_{S \subset \{1,2,\ldots,N\}} \prod_{i\in S} \alpha_i \left\langle \prod_{i \in S} \alpha_i \right\rangle^\beta \\ &= P_\beta 2^{-N} \sum_{S \subset \{1,2,\ldots,N\}} \frac{1}{2P_\beta} \left\{ \prod_{i\in S}\alpha_i \left\langle \prod_{i\in S} \alpha_i \right\rangle + \beta \prod_{i \in S} \alpha_i \left\langle \beta \prod_{i\in S} \alpha_i \right\rangle \right\} \\ &= 2^{-N-1} \sum_{S \subset \{1,2,\ldots,N,N+1\}} \prod_{i\in S}\alpha_i \left\langle \prod_{i \in S} \alpha_i \right\rangle \end{aligned}$$

Q.E.D. In the second equality we used Eq. (A3).

The expression (A2) for $P_{\vec{\alpha}}$ can be written

$$P_{\vec{\alpha}} = \left\langle \prod_i \frac{1+\alpha_i\alpha'_i}{2} \right\rangle_{\alpha'} \ . \qquad \text{(A4)}$$

$\langle\ \rangle_{\alpha'}$ denotes the average over the α'_i assuming for $\vec{\alpha}$ and $\vec{\alpha}'$ the same probability distribution. If all correlations are equal to the product of the mean values

$$\left\langle \prod_i \alpha_i \right\rangle = \prod_i \langle \alpha_i \rangle \ ,$$

we can take the product out of the average in Eq. (A4)

$$P_{\vec{\alpha}} = \prod_i \left\langle \frac{1+\alpha_i\alpha'_i}{2} \right\rangle_{\alpha'} = \prod_i P_{\alpha_i} \ . \qquad \text{(A5)}$$

This last equation expresses statistical independence of the N spins. Hence statistical independence of the N spins is equivalent to the fact that all correlations are equal to the product of the mean values. The redundancy

$$R[\vec{\alpha}] = \left\langle \ln \frac{P_{\vec{\alpha}}}{\prod_i P_{\alpha_i}} \right\rangle$$

measures the difference of $P_{\vec{\alpha}}$ and $\prod_i P_{\alpha_i}$ and it is standard information theory that $R[\vec{\alpha}] \geq 0$ and that $R[\vec{\alpha}]=0$ is equivalent to $P_{\vec{\alpha}} = \prod_i P_{\alpha_i}$.

We summarize by applying these results to the case of two spins. It is easy to prove now that we can express statistical independence of two spins by any of the four following equivalent relations:

$$P_{\alpha\beta} = P_\alpha P_\beta, \quad M[\alpha;\beta] \equiv -\left\langle \ln \frac{P_{\alpha\beta}}{P_\alpha P_\beta} \right\rangle = 0,$$

$$\langle \alpha\beta \rangle = \langle \alpha \rangle \langle \beta \rangle, \quad \langle \alpha \rangle^\beta = \langle \alpha \rangle \ .$$

[1] R. Linsker, Computer **21**, 105 (1988).

[2] R. Linsker, Neural Comput. **4**, 691 (1992).

[3] J. Rubner and P. Tavan, Europhys. Lett. **10**, 693 (1989).

[4] J. Rubner and K. Schulten, Biol. Cybern. **62**, 193 (1990).

[5] M. D. Plumbley, Neural Networks **6**, 823 (1993).

[6] H. Kühnel and P. Tavan, in *Parallel Processing in Neural Systems and Computers,* edited by R. Eckmiller, G. Hartmann, and G. Hauske (Elsevier Science, Amsterdam, 1990), pp. 187–190.

[7] R. Linsker, Neural Comput. **1**, 402 (1989).

[8] H. G. Schuster, Phys. Rev. A **46**, 2131 (1992).

[9] N. Redlich, Neural Comput. **5**, 750 (1993).

[10] B. A. Pearlmutter and G. E. Hinton, in *Proceedings of Neural Networks for Computing,* edited by S. J. Denker (American Institute of Physics, New York, 1986).

[11] G. Deco and L. Parra, Network (to be published).

[12] P. Földiak, Biol. Cybern. **64**, 165 (1990).

[13] J. Atick and N. Redlich, Neural Comput. **2**, 308 (1990).

[14] J. Atick and N. Redlich, Neural Comput. **4**, 196 (1992).

[15] C. Peterson and J. R. Anderson, Complex Syst. **1**, 995 (1987).

[16] C. Peterson and E. Hartman, Neural Networks **2**, 475 (1989).

[17] G. Parisi, *Statistical Field Theory* (Addison-Wesley, Reading, MA, 1988).

[18] J. Hertz, A. Krogh, and R. G. Palmer, *Introduction to the Theory of Neural Computation* (Addison-Wesley, Reading, MA, 1991).

[19] J. J. Hopfield, Proc. Nat. Acad. Sci. U.S.A. **81**, 3088 (1984).

[20] B. Schürmann, Phys. Rev. A **40**, 2681 (1989).

[21] E. Mach, I. Akad. Wiss. Wien Sitzungsber. Math. Nat. Cl. **52:2**, 303 (1865).

[22] G. E. Hinton, Neural Comput. **1**, 143 (1989).

[23] C. C. Galland, Network **4**, 355 (1993).

[24] A. L. Yuille, Neural Comput. **6**, 341 (1994).

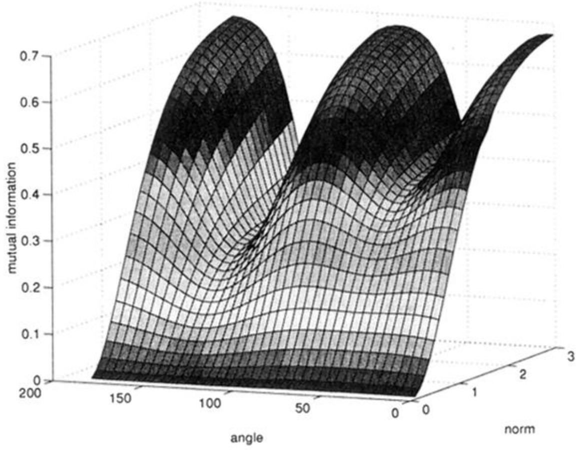

FIG. 3. Mutual information (bits) for a single output neuron plotted over the angle to the x axis (deg) and the norm of the weight $\vec{f}$. The input distribution on $\{\pm 1\}^2$ is described in the text.