Pergamon

Neural Networks, Vol. 10, No. 4, pp. 683–691, 1997
© 1997 Elsevier Science Ltd. All rights reserved
Printed in Great Britain
0893–6080/97 $17.00+.00

PII: S0893-6080(96)00110-4

CONTRIBUTED ARTICLE

Non-linear Feature Extraction by Redundancy Reduction in an Unsupervised Stochastic Neural Network

G. DECO AND L. PARRA*

Siemens AG

(*Received* 24 *February* 1995; *accepted* 10 *July* 1996)

Abstract—*Unsupervised feature extraction by a stochastic neural network can be defined as a minimization of the redundancy between the elements of the output layer, given complete information transfer from input to output. Redundancy minimization can be achieved by minimization of the mutual information between the units of the output layer. Complete information transfer is enforced by maximizing the mutual information of the input and output. With these two conditions we define a novel learning algorithm for stochastic recurrent networks. The minimum of redundancy corresponds to the extraction of statistically independent features, leading to a factorial representation of the environment. The resulting learning rule includes Hebbian and anti-Hebbian learning terms. These two terms are weighted by the amount of information transmitted in the learning synapse minus the grade of redundant information in the corresponding output neuron, giving thus, an information-theoretic interpretation of the proportionality constant of Hebb's biological rule. Simulations demonstrate the performance of this method. When a retina is simulated, the learning algorithm forms decorrelated receptive fields. This represents the first experiment that extends the results of the linear principle component analysis to the nonlinear case by a direct implementation of Barlow's principle of redundancy reduction for unsupervised features extraction by receptive fields formation in a retina model.* © *1997 Elsevier Science Ltd.*

Keywords—Boltzmann machine, Redundancy, Factorial codes.

1. INTRODUCTION

One of the most important tasks of cognition is to detect "statistical coincidences" in any combination of sensory stimuli. As indicated by Barlow (1989), a cognitive system needs to know whether a combination of sensory input (a sensory event) is an expected or an unexpected event. The brain tries to find causal relationships beween the sensory environment and the motor actions and consequences it plans to take. A well known principle introduced among others, by Zipf (1949) and by Attneave (1954) states, that the nervous system may be preprocessing the information of its sensory inputs in order to extract statistically independent features. In his seminal work Barlow (1989) related this problem with the principal objective of unsupervised learning. A perceptual system generates internal state representation of an unknown environment to represent external events. The aim of unsupervised learning is to find a set of features (symbols) to represent the messages such that the occurrence of each feature is independent of the occurrence of any other, i.e., that the extracted classes or features are non-redundant. This kind of learning is called factorial learning. This means that the joint probability of the internal states can be decomposed in the product of statistically independent probabilities. The result of the unsupervised learning described above is an internal factorial code that represents the sensorial inputs.

At the same time Linsker (1988, 1989, 1992) proposed the well known concept, called ''infomax'', derived from information theory. According to it, synaptic weights adapt in a constrained fashion in order to maximize mutual information between input and output layers of a cortical network. Atick and Redlich (1990) demonstrated that statistically salient input features can be optimally extracted from a noisy input by maximizing mutual information. The works of Redlich (1993a, b), and Atick and Redlich (1992) concentrate on the original idea of unsupervised learning of Barlow where feature extraction is handled as redundancy reduction. Some algorithms were developed for maximizing mutual information by using probabilistic linear neurons (Linsker, 1992) or nonlinear neurons (Linsker, 1989; Becker,

Requests for reprints should be sent to G. Deco, Siemens AG, Corporate Research and Development, ZFE T SN 4 Otto-Hahn-Ring 6, 81739, Munich, Germany; E-mail: gustavo.deco@zfe.siemens.de).

* Also at the Ludwig-Maximilian-Universität, lnstitut für Medizitffsche Oprik, Munich, Germany.

1992). In the linear case the infomax principle is related to the principal component analysis when deterministic networks are used (no noise on the output) and the covariance of the input noise is a diagonal matrix (Földiak, 1989). Rubner and Tavan (1989), and Földiak (1989) proved that a network composed of linear neurons can be trained to perform the principal component analysis, if the synaptic adaptation is defined as Hebbian in the vertical sense, i.e., from input to outputs, and anti-Hebb for the inhibitory lateral synaptic connections between the output units. The Hebbian and anti-Hebbian form of the learning rules can be derived from first principles using Information theoretic concepts (Linsker, 1988; Kuehnel & Tavan, 1990).

In this work we define a learning paradigm for a recurrent stochastic neural network that performs nonlinear and factorial feature extraction. By maximizing the mutual information between sensory inputs and output of the network we generate an internal representation without loss of information. At the same time the mutual information between the output neurons is explicitly minimized in order to eliminate the redundancy between the extracted features. The result leads to a factorial code that represent the sensory events without loss of information. In this way we extend the unsupervised learning principle of Barlow for probabilistic nonlinear neurons and for networks which include recurrences. The learning algorithm can be interpreted as a weighted combination of Hebbian and anti-Hebbian rule.

The weighting term is the amount of information transmitted in the learning synapse minus the grade of generated redundancy, giving thus an information-theoretic interpretation of the constant of Hebb's biological rule. Correlated neurons will reinforce the synapses iff for these channels no information loss and no redundancy is introduced. We demonstrate the performance of the model on five different standard experiments where a factorial code can be found. We also simulate a simple retina model, in which case the learning algorithm forms decorrelated receptive fields, that extract statistical independent features from the retina. This gives us an interpretation and mechanism of formation of receptive fields in the visual cortex (see also Rubner & Schulten, 1990; White, 1992).

2. THEORETICAL FORMULATION

The present model is based on a stochastic neural network architecture with separated input and output units. The basic concept for this model was first introduced as a Boltzmann machine by Ackley et al. (1985). We ignore possible hidden neurons for the sake of simple presentation. (The extension of the formalism for hidden neurons is straightforward). Let us label the state of the output units by α. Fig. 1 represents a network without hidden units. The Boltzmann–Gibbs distribution of the states of the output neurons for a fixed input pattern γ can be written as,

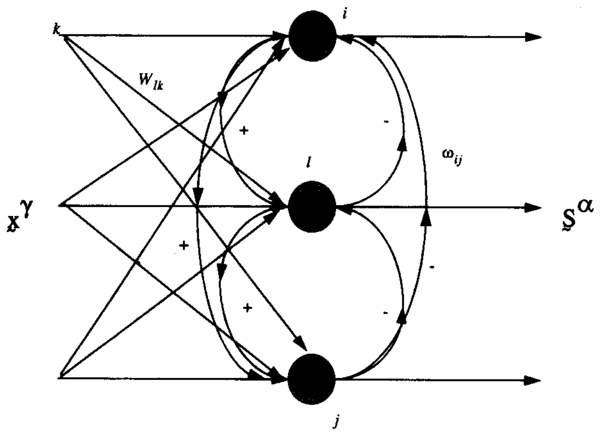

FIGURE 1. Stochastic neural architecture with direct and lateral synaptic connections.

$$P_{\alpha/\gamma} = \frac{e^{-\beta E^{\alpha/\gamma}\beta}}{\sum\limits_{\alpha} e^{-\beta E^{\alpha/\gamma}}} \tag{1}$$

where $E^{\alpha/\gamma}$ is the energy function,

$$E^{\alpha/\gamma} = -\frac{1}{2}\sum_{ij} w_{ij}\cdot S_j^{\alpha}\cdot S_j^{\alpha} - \frac{1}{2}\sum_{ij} W_{ij}\cdot X_i^{\gamma}\cdot S_j^{\alpha} \tag{2}$$

In eqn (1), $P_{\alpha/\gamma}$ denotes the conditional probability of the possible configurations α of the output neurons given pattern γ at the input. In eqn (2) S_i^{α} denotes the value of the output neuron "i" for the output configuration α. The parameter β in eqn (1) is related to the inverse of the temperature. If the connections w_{ij} between the neurons are symmetric, than an energy function can be defined and the statistical mechanics Boltzmann–Gibbs distribution gives the probability of finding the system in a determined state S given the external fields h_i^{ext}. Asymmetric synapses w_{ij} in Lyapunov functions might be used as well. This case was described thoroughly by Schürmann (1989). The synapses W_{ij} connect the external input vector **X** with the net and don't have to be symmetric. The external input vectors $\mathbf{X}^{\gamma}$ assume real values that may be drawn from a given probability distribution P_{γ}, with γ labeling the input patterns. We point out that the input units should not be understood as Ising spins. For a concrete γ-pattern they are fixed, and determines a fixed external field.

The probability P_i that the neuron i is in state 1 can be defined in terms of the probabilities $P_{\alpha/\gamma}$ and P_{γ} as,

$$P_i = \sum_{\alpha} S_i^{\alpha} P_{\alpha} = \sum_{\gamma,\alpha} S_i^{\alpha} P_{\alpha/\gamma} P_{\gamma} \tag{3}$$

In order to implement factorial learning, at first the information should be transferred to the output neurons. Secondly the generated probability distribution of the individual output neurons should be statistically independent.

The information theoretic concept we are suggesting involves the measurement of entropies. These can all be expressed in terms of the probabilities $P_{\alpha/\gamma}$ and P_γ. Let us define the entropy $H(\alpha/\gamma)$ of the joint distribution of the outputs for a fixed input γ and the entropy $H(\alpha)$ of the joint distribution of the outputs independent of a special input,

$$H(\alpha/\gamma) = -\sum_\gamma P_\gamma \sum_\alpha P_{\alpha/\gamma} log P_{\alpha/\gamma} \tag{4}$$

$$H(\alpha) = -\sum_\alpha P_\alpha log P_\alpha$$

$$= -\sum_\gamma P_\gamma \sum_\alpha P_{\alpha/\gamma} \log\left(\sum_\gamma P_\gamma P_{\alpha/\gamma}\right) \tag{5}$$

The first objective of learning will be achieved by maximizing the transfer of the information from the input to the output layer. A general measure of the transmitted information introduced by Shannon (1948) is the mutual infomation (MI), which can be expressed in terms of the defined entropies. We refer to the mutual information between input and output as "vertical" mutual information (*MIV*). It can be written as

$$MIV = H(\alpha) + H(\gamma) - H(\alpha,\gamma) = -H(\alpha/\gamma) + H(\alpha) \tag{6}$$

$$= \sum_\gamma P_\gamma \sum_\alpha P_{\alpha/\gamma} \log(P_{\alpha/\gamma})$$

$$-\sum_\gamma P_\gamma \sum_\alpha P_{\alpha/\gamma} \log\left(\sum_\gamma P_\gamma P_{\alpha/\gamma}\right) \tag{7}$$

The mutual information is always positive and the maximum value of *MIV* is $H(\gamma)$, i.e.

$$0 \le MIV \le H(\gamma) \tag{8}$$

The second objective of learning is to generate a factorial output code, i.e., to extract statistically independent features. This implies that the occurrence of each symbol, i.e., each active output unit, is independent of the occurrence of any other. Statistical independence among the binary output units can be expressed by the condition

$$P_\alpha = \prod_i P(S_i = S_i^\alpha) = \prod_i (S_i^\alpha P_i + (1 - S_i^\alpha)(1 - P_i)) \tag{9}$$

As pointed out by Atick and Redlich (1992), factorial code is equivalent to a vanishing mutual information between the output units. This mutual information will be labeled "horizontal" (*MIH*). It is defined as the difference between the sum of the entropy $H(j)$ of each output "j" and the entropy of the joint output states $H(\alpha)$,

$$MIH = \sum_j H(j) - H(\alpha) \ge 0 \tag{10}$$

The sum over the single neuron entropy $H(j)$ is usually called "bit entropy" and is defined by,

$$\sum_j H(j) = \sum_j S_j^\alpha P_j + (1 - S_j^\alpha)\log(1 - P_j) \tag{11}$$

The identity in eqn (10) is equivalent to eqn (9). This means that a factorial code (eqn (9)) is obtained by minimizing the horizontal mutual information, which is a measure of the grade of "dependency" among the outputs. The unsupervised learning that we introduce in this paper for a stochastic network described by eqn (1) consists in maximizing *MIV* up to its upper bound $H(\gamma)$ and minimizing *MIH*. We therefore choose the following cost function,

$$C = (H(\gamma) - MIV) + MIH \tag{12}$$

$$= \sum_j H(j) + H(\alpha,\gamma) - 2 \cdot H(\alpha) \tag{13}$$

Here the entropy $H(\alpha,\gamma)$ of the joint probability distribution of the input data and output neuron states is given by

$$H(\alpha,\gamma) = -\sum_{\gamma,\alpha} P_{\alpha\gamma} log P_{\alpha\gamma} = -\sum_{\gamma,\alpha} P_\gamma P_{\alpha/\gamma} \log(P_\gamma P_{\alpha/\gamma}) \tag{14}$$

In order to minimize C we perform gradient descendent corrections of the weights. This yields the following learning rule

$$w_{ij}^{new} = w_{ij}^{old} - \eta \cdot \frac{\partial C}{\partial w_{ij}};\ W_{ij}^{new} = W_{ij}^{old} - \eta \cdot \frac{\partial C}{\partial w_{ij}} \tag{15}$$

where η is a learning constant. In the following learning rules are derived

$$\Delta w_{ij} = \eta \frac{\beta}{2} \sum_{\gamma,\alpha} P_{\alpha/\gamma} P_\gamma \times \left(\log\left(\frac{P_{\alpha/\gamma}}{P_\alpha}\right) - \log\left(\frac{P_\alpha}{\prod_l (s_l^\alpha P_l + (1 - s_l^\alpha)(1 - P_l))}\right)\right) \times (s_i^\alpha s_j^\alpha - \langle s_i s_j \rangle_\gamma) \tag{16}$$

$$\Delta w_{ij} = \eta \frac{\beta}{2} \sum_{\gamma,\alpha} P_{\alpha/\gamma} P_\gamma \times \left(\log\left(\frac{P_{\alpha/\gamma}}{P_\alpha}\right) - \log\left(\frac{P_\alpha}{\prod_l (s_l^\alpha P_l + (1 - s_l^\alpha)(1 - P_l))}\right)\right) \times (s_i^\alpha x_j^\gamma - \langle s_i x_j^\gamma \rangle_\gamma) \tag{17}$$

The interpretation of the obtained unsupervised learning rule is interesting. A Hebbian term is given by $S_i^\alpha S_j^\alpha$ in eqn (16) representing the instantaneous correlation between the neurons. The term $-\langle S_i S_j \rangle_\gamma$ is an anti-Hebbian term given by the averaged correlation between the neurons. These terms are the weighted sum over all possible states, where the weighting factor is the difference between the information transmitted from input to output and the redundancy in the

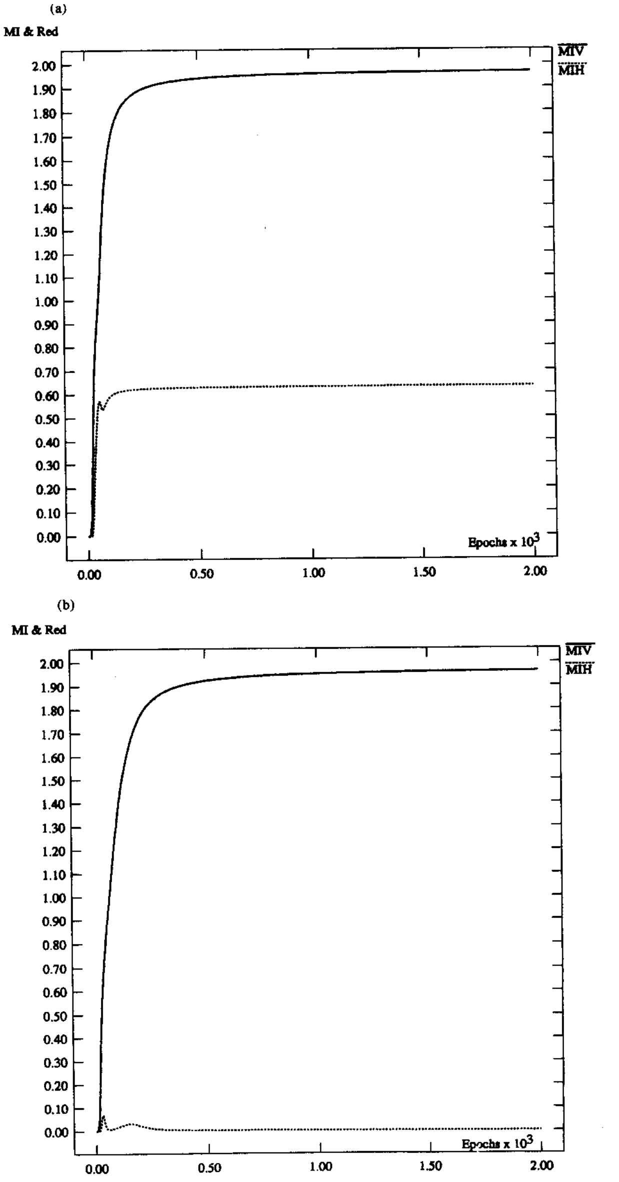

FIGURE 2. (a) Evolution during learning of experiment 1 of the vertical and horizontal mutual information (*MIV*) and (*MIH*) when unsupervised learning includes only maximization of the transmission of information. (b) Evolution during learning of experiment 1 of the vertical and horizontal mutual information (*MIV*) and (*MIH*) when unsupervised learning includes maximization of the transmission of information with minimization of redundance.

output layer. In other words, the channels that transmit information without giving redundance at the output (factorial code) will be reinforced with the Hebbian rule. The anti-Hebbian term is weighted with the same factor and takes the average over all possible channels into account, representing that way a forgetting factor. We obtain a weighted correction according to whether the activation of the postsynaptic cell exceeds its average value. This holds also true for the connections between input and neurons.

TABLE 1
Nonuniform Distributed Patterns (Exp. 1)

P_γ	Code I	Code F-a	Code F-b
0.2	0 0 1	0 0 1	1 0 1
0.3	0 1 1	0 1 1	1 1 1
0.3	1 1 0	0 1 0	0 1 0
0.2	1 1 1	1 1 0	1 1 0

3. EXPERIMENTAL AND RESULTS

3.1. Implementation of the Learning Rule

We have implemented the learning rule given by eqns (16) and (17). This learning rule requires the calculations of the probability $P_{\alpha/\gamma}$ with eqns (2)–(4) and of the probability P_α by summing $P_{\alpha/\gamma}$ over the training patterns. We calculate this probabilities for all possible α, i.e., for the 2^n states of the output layer with n output neurons. Now let us analyse the complexity of the algorithm that performs the unsupervised learning. Assume, the input has d dimensions and the number of training patterns is N. The complexity of the algorithm that calculates an update for all weights is $O(d \times n \times N \times 2^n)$, which is the same as in an equivalent Boltzmann machine.

In the next section we demonstrate some applications of the unsupervised redundancy reduction learning with the presented stochastic network. We apply the learning paradigm to different benchmark problems for factorial learning defined in the literature (see Barlow et al., 1989; Hentschel & Barlow, 1991; Schmidhuber, 1992).

3.2. Binary Coding and Compression Experiments

The goal of the following examples is to find a nonlinear invertible transformation of a binary input code I in a factorial output code F. The invertability is assured by perfect transmission of input information into the new code F. We will consider non-uniform distributions P_γ of the input code I, since these distributions showed empirically to be more challenging when used in re-coding and compression tasks. We differentiate between a local and a distributed code. In a local representation each input pattern has only one non-zero bit. In the distributed representation the code is distributed on different active bits. Therefore the distributed representation can code in a d-dimensional vector 2^a different binary inputs. In our examples both representations will be considered.

TABLE 2
Nonuniform Distributed Patterns (Exp. 1)

Code	Bit entropy	*MIH*	*MIV*	*R* (%)
I	2.85	–	–	45
F-a	2.60	0.63	1.97	32
F-b	1.97	0.00	1.97	0

TABLE 3
Nonuniform Distributed Patterns (Exp. 2)

P_γ	Code I	Code F-a	Code F-b
1/9	0 0 0 1	1 0	1 1
2/9	0 0 1 0	1 1	0 1
2/9	0 1 0 0	0 1	1 0
5/9	1 0 0 0	0 0	0 0

In the following subsections we will use the definition of redundancy as given by Barlow et al. (1989). It is based on the definition of the mutual information, that we called horizontal, given in eqn (10). The redundancy R is the horizontal mutual information normalized by the entropy of the joint distribution. It measures how far a given code diverges from a factorial representation. Here it is given for the distribution of the input coordinates.

$$R = \frac{\sum_j H(j) - H(\gamma)}{H(\gamma)} \tag{3.3}$$

3.2.1. *Experiments 1 and 2: Non-uniform Distributed Input Patterns.* In the first example the input code I consists in four patterns with nonuniform input distribution in a distributed representation (see Table 1). Three input dimension and three output dimension are used. The input bit entropy is 2.85 and the entropy of the patterns distribution is $H(\gamma) = 1.97$. Fig. 2a shows the evolution of the vertical and horizontal mutual information, where unsupervised learning includes only maximization of the transmission of information. Fig. 2b shows the same for the case, where the unsupervised learning rule includes maximization of the transmission of information with simultaneous minimization of redundance. In both cases *MIV* is maximized reaching its maximum possible value $H(\gamma) = 1.97$, and yielding, therefore, a revertible code. However, in the first case the redundancy is not taken into account, and therefore, the horizontal mutual information increases during learning yielding a redundant code F-a (see Fig. 2a). In the second case eqn (16) is used, i.e., *MIV* is maximized and *MIH* is minimized yielding an invertible and factorial code F-b. The values of the entropies for code I and the generated codes F-a and F-b after training are presented in Table 2.

Table 1 shows the input code I, its distribution, the invertible factorial code F-a corresponding to Fig. 2a, and the invertible but non-factorial code F-b corresponding to Fig. 2b.

TABLE 4
Nonuniform Distributed Patterns (Exp. 2)

Code	Bit entropy	*MIH*	*MIV*	*R* (%)
I	3.03	–	–	65
F-a	1.91	0.06	1.84	3.8
F-b	1.84	0.00	1.84	0

TABLE 5
Coding Geometric Progressions (Exp. 3)

Code	Bit entropy	*MIH*	*MIV*	*R* (%)
I	4.33	–	–	45
F	2.99	0.0	2.99	0

The following is a specific example used also by Schmidhuber (1992), where the input patterns are nonuniformly distributed. It is an interesting case, since only one factorial code (F-b in Table 3) and many other invertible codes with low redundancy exist here. The results are summarized in Tables 3 and 4. The codes F-a and F-b are 2-dimensional and correspond to the results of unsupervised learning by maximizing *MIV* without and with minimization of *MIH* respectively. The input entropy of the patterns is $H(\gamma) = 1.84$.

For the input code I we have used a 4-dimensional, redundant, local distribution.

3.2.2. *Experiment 3: Coding Geometric Progressions.* It is possible to show that geometric progressions have an exact factorial representation consisting of a binary

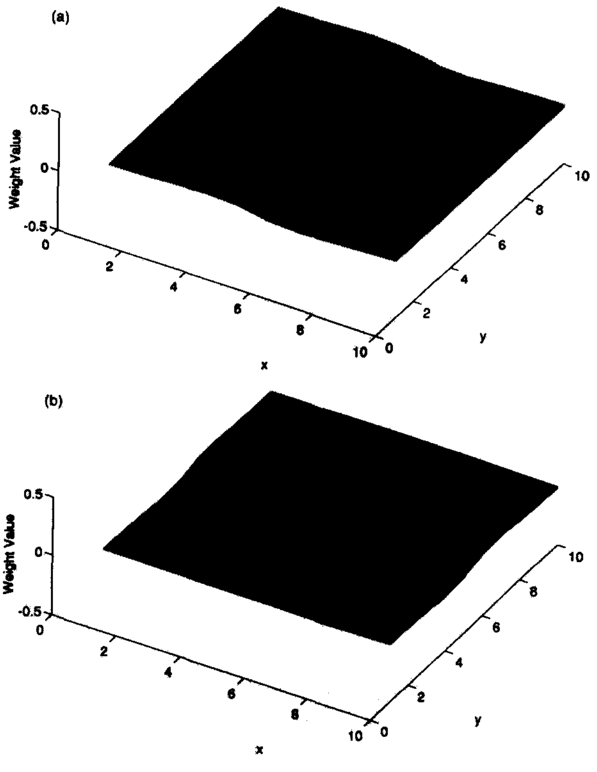

FIGURE 3. Receptive field formed by neuron output 1 (a) or neuron output 2 (b) by unsupervised nonlinear features extraction after training with random Gaussian spots as sensory inputs on a retina.

TABLE 6
Coding Geometric Progression (Exp. 3)

Code I	Code F
0 0 0 0 0 0 0 1	0 1 0
0 0 0 0 0 0 1 0	1 1 1
0 0 0 0 0 1 0 0	0 1 1
0 0 0 0 1 0 0 0	0 0 1
0 0 0 1 0 0 0 0	0 0 0
0 0 1 0 0 0 0 0	1 0 1
0 1 0 0 0 0 0 0	1 1 0
1 0 0 0 0 0 0 0	1 0 0

sequence coding (see Barlow et al., 1989; Hentschel & Barlow, 1991). We show in this section the results obtained using a input local representation of an 8-dimensional input forming a geometric progression $P_\gamma = Kx^\gamma$ with $x = 0.95$ and K being a proper normalization constant. The output layer has three neurons. Table 5 shows the results after training according to eqn (17). The code F is the invertible factorial code obtained after training.

The code F is explicitly given in Table 6.

3.2.3. *Experiment 4: Power-law Coding.* As remarked by Hentschel and Barlow (1991) another important distribution is the power-law distribution $P_\gamma = K_\gamma^{-L}$ with K being a proper normalization constant. This kind of distribution is of interest, since for example the word distribution in normal English vocabulary follows this power-law distribution. The output layer has two neurons and the input code has again a local 4-dimensional code. All the results for the obtained factorial code are presented in Tables 7 and 8. The code F-a resulted after maximization of *MIV* only. The code F-b was obtained by simultaneously reducing redundancy. In fact code F-b is factorial and invertible and code F-a is invertible but not factorial.

3.3. Receptive Fields Formation from a Retina

As remarked by Rubner and Schulten (1990) the processing of spatial information has, for a long time, been a problematic issue. Hubel and Wiesel (1962) described the first cells as feature detectors (cells in the cat primary visual cortex, area 17). These neurons perform the first stage of spatial information processing in the primary

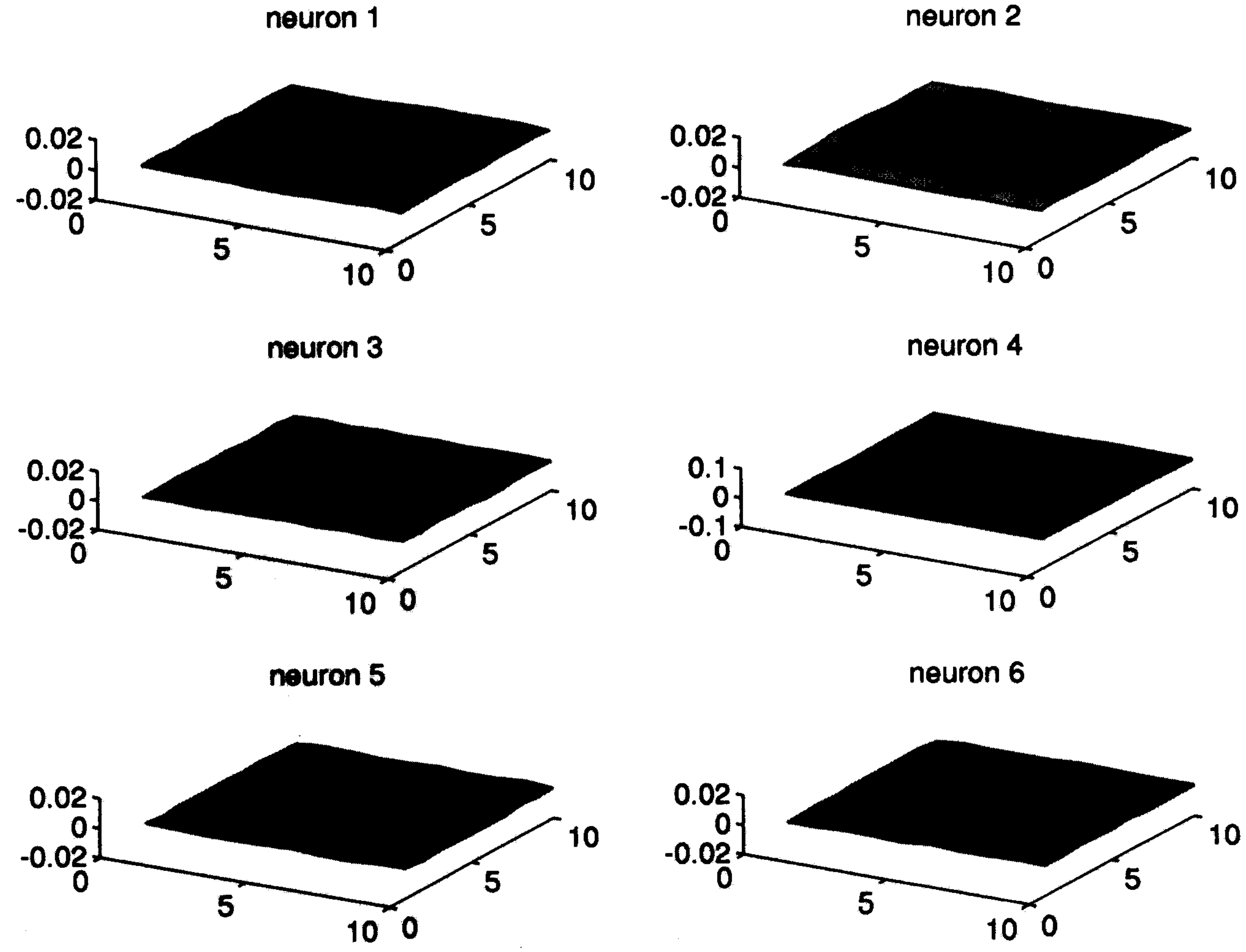

FIGURE 4. Receptive fields formed by a layer of six output neurons.

TABLE 7
Power-law Coding (Exp. 4)

Code	Bit entropy	*MIH*	*MIV*	*R* (%)
I	2.20	–	–	45
F-a	1.35	0.068	1.28	5.47
F-b	1.28	0.0	1.28	0

TABLE 8
Power-law Coding (Exp. 4)

Code I	Code F-b
0 0 0 1	0 1
0 0 1 0	1 1
0 1 0 0	0 0
1 0 0 0	1 0

visual cortex. The question is whether spatial information in the visual cortex is processed based on the extraction of local features, or on a Fourier-like decomposition into spatial-frequency channels (see Campbell & Robson, 1968; MacKay, 1981; Pollen et al., 1971). Rubner and Schulten (1990) describe a mechanism of formation of spatial feature detectors for the case of neurons with linear response. This is the result of the classical principal component analysis. The goal of this section is to study the formation of spatial detectors for the case of nonlinear features extraction by applying the learning paradigm introduced in this paper.

We show how the present learning paradigm is able to form receptive fields in an input-retina. The simple retina model consists of an array of 10 × 10, that is a total of 100 input neurons and two output neurons. Each input vector was a Gaussian spot centered at a random position at least two input units away from the nearest edge in the input array. Redundancy reduction causes a de-correlation of the synapses during learning. In other words, the synapses are reinforced in order to extract information of the input-screen but only if this leads to a non-redundant representation. In this case, the non-redundancy is assured by the formation of receptive field in the synapse space. This mechanism give us an information-theoretic first principle for the formation of receptive fields in a retina.

Fig. 3a and b show the resulting values of the synapses which connect the retina to the first and second output neuron respectively. The x- and y-axis indicate the coordinates on the input lattice (retina) and the z-axis is the value of the respective synapse. From these figures it is easy to see, that the two generated receptive fields divide the input space in four equal regions, corresponding to the four output codes: 1 1 (two overlapping hills), 1 0 (output neuron 1 valley overlapping with output neuron 2 hill), 0 1 (output neuron 1 hill overlapping with output neuron 2 valley) and 0 0 (two overlapping valleys). The division of the retina in four spatially different sections

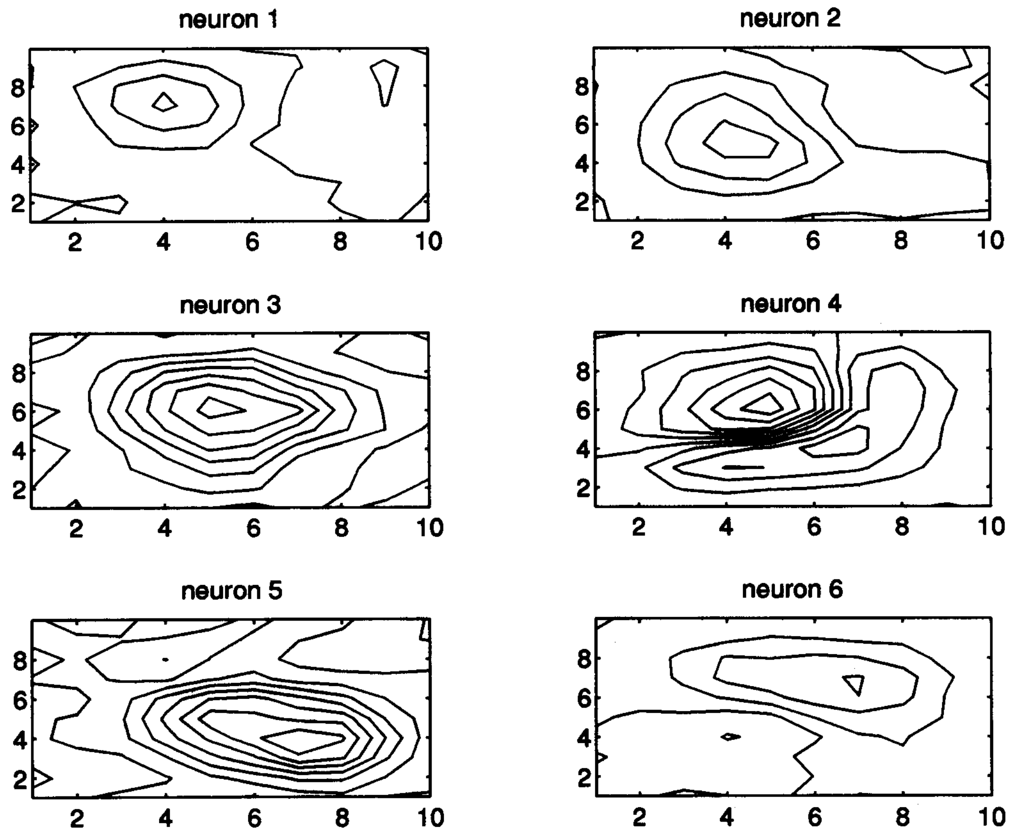

FIGURE 5. Contours of receptive fields formed by a layer of six output neurons.

maximizes the extracted information by simultaneously de-correlating the response activation of the two neurons. Already with two neurons we find the formation of spatial selective cells.

A quite interesting result is observed when the output layer contains six neurons. The formed receptive fields are shown in Figs 4 and 5. As in the linear case analysed by Rubner and Schulten (1990) our learning paradigm yields mutually orthogonal, spatially oscillating receptive fields. Receptive fields of simple cells in cat striate cortex exhibit the same oscillatory patterns (Jones & Palmer, 1987a; Jones et al., 1987b). The receptive fields formed display excitatory and inhibitory regions and reflect simple cells, that respond selectively to edges or bars of fixed orientation. Therefore, following the redundancy reduction principle of Barlow given complete information transfer, the weights converge to form detectors of mutually independent features of the environment, which represents a nonlinear generalization of the linear principal component analysis networks (Rubner & Schulten, 1990).

REFERENCES

Ackley, D., Hinton, G., & Sejnowski (1985). A learning algorithm for Boltzmann machines. *Cognitive Science*, **9**, 147–169.

Atick, J., & Redlich, A. (1990). Towards a theory of early visual processing. *Neural Computation*, **2**, 308–320.

Atick, J., & Redlich, A. (1992). What does the retina know about natural scenes. *Neural Computation*, **4**, 196–210.

Attneave, F. (1954). Informational aspects of visual perception. *Psychological Review*, **61**, 183–193.

Barlow, H. (1989). Unsupervised learning. *Neural Computation*, **1**, 295–311.

Barlow, H., Kaushal, T., & Mitchison, G. (1989). Finding minimum entropy codes. *Neural Computation*, **1**, 412–423.

Becker, S. (1992). *An Information-theoretic Unsupervised Learning Algorithm for Neural Networks*. unpublished doctoral thesis, University of Toronto.

Campbell, F., & Robson, I. (1968). Application of Fourier analysis to the visibility of gratings. *J. Physiol. (London)*, **197**, 551–566.

Földiak, P. (1989). Adaptive network for optimal linear feature extraction. In *Proceedings of the IEEE/INNS International Joint Conference on Neural Networks, Washington DC, Vol. 1* (pp. 401–405). New York; IEEE Press, Publisher NY.

Hentschel, H.G.E., & Barlow, H.B. (1991). Minimum-entropy coding with Hopfield networks. *Network*, **2**, 135–148.

Hubel, D., & Wiesel, T. (1962). Receptive Fields, binocular interaction and functional architecture in the cat's visual cortex. *Journal of Physiology (London)*, **160**, 106–154.

Jones, J.P., & Palmer, L.A. (1987). The two-dimensional spatial structure of simple receptive fields in cat striate cortex. *Journal of Neurophysiology*, **58**, 1187–1211.

Jones, J.P., Stepnoski, A., & Palmer, L.A. (1987). The two-dimensional spectral structure of simple receptive fields in cat striate cortex. *Journal of Neurophysiology*, **58**, 1112–1232.

Kuehnel, H., & Tavan, P. (1990), The anti-Hebb rule derived from information theory. In R. Eckmiller, G. Hartmann & G. Hauske (Eds.), *Parallel processing in neural systems and computers*, (pp. 187–190). North-Holland: Elsevier Science.

Linsker, R. (1988). Self-organization in a perceptual network. *Computer*, **21**, 105.

Linsker, R. (1989). How to generate ordered maps by maximizing the mutual information between input and output signals. *Neural Computation*, **1**, 402–411.

Linsker, R. (1992). Local synaptic learning rules suffice to maximize mutual information in a linear network. *Neural Computation*, **4**, 691–702.

MacKay, D. (1981). Strife over visual cortical functions. *Nature*, **289**, 117–118.

Pollen, D., Lee, J., & Taylor, I. (1971). How does the striate cortex begin the reconstruction of the visual world? *Science*, **173**, 74–77.

Redlich, A.N. (1993). Redundancy reduction as a strategy for unsupervised learning. *Neural Computation*, **5**, 289–304.

Redlich, A.N. (1993). Supervised factorial learning. *Neural Computation*, **5**, 750–766.

Rubner, J., & Tavan, P. (1989). A self-organization network for principal-component analysis. *Europhysics Letters*, **10**, 693–698.

Rubner, J., & Schulten, K. (1990). Development of feature detectors by self-organization. *Biological Cybernetics*, **62**, 193–199.

Schmidhuber, J. (1992). Learning factorial codes by predictability minimization. *Neural Computation*, **4**, 863–879.

Schürmann, B. (1989). Stability and adaptation in artificial neural systems. *Physical Review A*, **40 (5)**, 2681–2688.

Shannon, C. (1948). A mathematical theory of communication. *Bell System Technical Journal*, **7**, 379–423.

White, R. (1992). Competitive Hebbian learning: algorithm and demonstrations. *Neural Networks*, **5**, 261–275.

Zipf, G. (1949). *Human behavior and the principle of least effort*. Cambridge, MA: Addison-Wesley.

APPENDIX

Here we perform the main calculations which lead to the learning rules discussed in . To calculate the derivatives in eqn (15) we depart from the definition of the objective function in eqn (13) and the definitions of the different entropies in eqns (11), (14) and (5), and we obtain

$$\frac{\partial C}{\partial w_{ij}} = -\sum\nolimits_{\gamma,\alpha} P_\gamma \frac{\partial}{\partial w_{ij}}(P_{\alpha/\gamma})(\log(P_\gamma P_{\alpha/\gamma})+1)$$
$$-\sum\nolimits_l \frac{\partial}{\partial w_{ij}}(P_l)(logP_l - \log(1-P_l) \qquad (A1)$$
$$+2\cdot\sum_{\gamma,\alpha} P_\gamma \frac{\partial}{\partial w_{ij}}(P_{\alpha/\gamma})(logP_\alpha + 1)$$

Using eqns (1) and (2) we obtain

$$\frac{\partial}{\partial w_{ij}}(P_{\alpha/\gamma}) + \frac{\beta}{2}P_{\alpha/\gamma}\left(S_i^\alpha S_j^\alpha - \sum_\alpha P_{\alpha'/\gamma} S_i^{\alpha'} S_j^{\alpha'}\right) \qquad (A2)$$

The derivative of $\partial(P_l)/\partial w_{ij}$ can be written according to eqn (3) as

$$\frac{\partial}{\partial w_{ij}}(P_l) = \sum_{\gamma,\alpha} P_\gamma S_l^\alpha \frac{\partial}{\partial w_{ij}}(P_{\alpha/\gamma}) \qquad (A3)$$

This gives

$$\frac{\partial C}{\partial w_{ij}} = -\sum_{\gamma,\alpha} P_\gamma \frac{\partial}{\partial w_{ij}} P_{\alpha/\gamma}\left(\log(P_\gamma P_{\alpha/\gamma}) + \sum_l S_l^\alpha(logP_l - \log(1-P_l))\right.$$
$$\left. - 2logP_\alpha - 1\right) \qquad (A4)$$

All summands in eqn (0) are of the form $\sum_{\gamma,\alpha} P_\gamma\, \partial(P_{\alpha/\gamma})/\partial w$ *Term*. We point out that if "*Term*" is independent of the states α, the Hebbian term in eqn (A2) is canceled by evaluating the sum over the states α. Thus, adding or substracting constant terms in the parenthesis of eqn (A4) is arbitrary. We use this to modify the expressions, as follows

$$\frac{\partial C}{\partial w_{ij}} = -\sum_{\gamma,\alpha} P_\gamma \frac{\partial}{\partial w_{ij}} P_{\alpha/\gamma}\left(logP_{\alpha/\gamma} + \sum_l (S_l^\alpha logP_l + (1-S_l^\alpha)\log(1-P_l))\right.$$
$$\left. - 2logP_\alpha\right) \qquad (A5)$$

Because $S \in \{0,1\}$ we can rewrite the second summand as

$$\sum_l (S_l^\alpha \mathit{log} P_l + (1 - S_l^\alpha)\log(1 - P_l)) = \log \prod_l (S_l^\alpha P_l + (1 - S_l^\alpha)(1 - P_l)) \tag{A6}$$

Introducing the symbol $\langle x \rangle_\gamma$ for the average value of x for a fixed pattern γ we can write the learning rule now as

$$\Delta w_{ij} = \eta \frac{\beta}{2} \sum_{\gamma,\alpha} P_{\alpha/\gamma} P_\gamma \times \left(\log\left(\frac{P_{\alpha/\gamma}}{P_\alpha}\right) - \log\left(\frac{P_\alpha}{\prod_l (s_l^\alpha P_l + (1 - s_l^\alpha)(1 - P_l))}\right)\right) \times (s_i^\alpha s_j^\alpha - \langle s_i s_j \rangle_\gamma) \tag{A7}$$

Mutatis mutandis, we obtain for the input connections the following learning rule

$$\Delta w_{ij} = \eta \frac{\beta}{2} \sum_{\gamma,\alpha} P_{\alpha/\gamma} P_\gamma \times \left(\log\left(\frac{P_{\alpha/\gamma}}{P_\alpha}\right) - \log\left(\frac{P_\alpha}{\prod_l (s_l^\alpha P_l + (1 - s_l^\alpha)(1 - P_l))}\right)\right) \times (s_i^\alpha x_j^\gamma - \langle s_i x_j^\gamma \rangle_\gamma) \tag{A8}$$