Proceedings of 1993 International Joint Conference on Neural Networks

Self-Organization in Stochastic Neural Networks

G. Deco and L. Parra
Siemens AG, Corporate Research and Development, ZFE ST SN 41
Otto-Hahn-Ring 6, 8000 Munich 83, Germany

Abstract

The maximization of the Mutual Information between the stochastic outputs neurons and the clamped inputs is used as unsupervised criterion for training a Boltzmann Machine. The resulting learning rule contains two terms corresponding to the Hebbian and anti-Hebbian learning. The two terms are weighted by the amount of transmitted information in the learning synapse, giving an information-theoretic interpretation to the proportionality constant given in the biological rule of Hebb.The anti-Hebbian term causes the convergence of weights. Simulation for the encoder problem demonstrates optimal performance of this method.

1.0 Introduction

Boltzmann Machines [1] are a class of stochastic neural networks which applied the principles of statistical mechanics to implement a kind of recurrent network with symmetrical connections which is capable to learn in a supervised fashion a given probability distribution. Boltzmann Machines can be seen as a generalization of the Hopfield networks by including hidden units. The learning algorithm derived for these stochastic networks relates in an efficient way the Boltzmann distributions with information theory for supervised training. On the other hand, Linsker [2-4] applied a well known concept from the information theory. He has proposed an optimization principle, called infomax, according to which synaptic weights develop in such a way that the mutual information between input and output layers of a cortical network is maximized under constrained boundary conditions. It has been proved that statistically salient input features can be optimally extracted from a noisy input by maximizing the mutual information. Some algorithms were developed for maximizing mutual information by using probabilistic linear neurons [4] or non-linear neurons in a probabilistic winner-take-all-network [3]. In the linear case the infomax principle is related to the Principal Component Analysis which is recovered when deterministic networks are used (noise of the output equal zero) and the covariance of the input noise is a diagonal matrix. The aim of the present work is to define for the Boltzmann Machine an unsupervised learning paradigm based on the maximization of the mutual information. In this way we extend the infomax principle for probabilistic non-linear neurons and for networks which include hidden neurons and recurrences. The learning algorithm yields an interesting weighted combination of Hebbian and anti-Hebbian rule. The weighted coefficients can be interpreted by the infomax principle. The algorithm is tested by using the encoder problem and optimal data compression is obtained by using the proposed algorithm.

2.0 Theoretical Formulation

Let as define a neural network composed by stochastic binary units S_i, taking output value $S_i = 1$ with probability p and value $S_i = -1$ with probability $1-p$. The probability p is given by,

$$p = \frac{1}{1+e^{(-2\tau\sum_j w_{ij}S_j)}} \tag{2.1}$$

Stochastic Units defined in this way describe the effect of thermal fluctuations in a system of Ising spins and is known as Glauber dynamic. The parameter τ in equation 2.1 is related with the inverse of the temperature. If the connections w_{ij} between the neurons are symmetric, than an energy function can be defined and the Boltzmann-Gibbs distribution from the statistical mechanics gives the probability of finding the system in a determined state $\{S\}$. Let us label the states of the input units by γ, of the output units by α and of the hidden units by β.Then the Boltzmann-Gibbs distribution of the states of the hidden and output neurons states for a fixed input pattern γ can be written as,

$$P_{\alpha\beta/\gamma} = \frac{e^{-\tau H^{\alpha\beta/\gamma}}}{Z_\gamma} \tag{2.2}$$

where Z_γ is the partition function and $H^{\alpha\beta/\gamma}$the energy function given by,

$$Z_\gamma = \sum_{\alpha\beta} e^{-\tau H^{\alpha\beta/\gamma}} \qquad H^{\alpha\beta/\gamma} = -\frac{1}{2}\sum_{ij} w_{ij}\cdot S_i^{\alpha\beta/\gamma}\cdot S_j^{\alpha\beta/\gamma} \tag{2.3}$$

The unsupervised learning that we introduce in this paper for a stochastic network described by equation 2.2 consists in maximizing the transfer of information from the input neurons to the output neuron. That means that a message γ coded in the input layer should be transmitted through the stochastic neurons so that the code given by the averaged thermal value of the output neurons contains the most information included in the original message γ. A measure of the transmitted information is given by the "Mutual Information" [5] that in our case can be written as,

$$M = \sum_\gamma P_\gamma \sum_\alpha P_{\alpha/\gamma}\log\left(P_{\alpha/\gamma}\right) - \sum_\gamma P_\gamma \sum_\alpha P_{\alpha/\gamma}\log\left(\sum_\gamma P_\gamma P_{\alpha/\gamma}\right) \tag{2.4}$$

where P_γ is the probability distribution of the input patterns and $P_{\alpha/\gamma}$ is the probability distribution of the possible configurations of the output neurons given that pattern γ is presented at the input (Conditional probability). In order to maximize the mutual information we perform gradient ascendent corrections on the weights. This yields following learning rule,

$$w_{ij}^{new} = w_{ij}^{old} + \eta\cdot\frac{\partial M}{\partial w_{ij}} \tag{2.5}$$

where η is a learning constant. The derivative in equation 2.5 can be calculated after some algebra,

$$\frac{\partial M}{\partial w_{ij}} = \sum_{\gamma} P_{\gamma} \sum_{\alpha,\beta} \frac{\partial}{\partial w_{ij}} (P_{\alpha\beta/\gamma}) \log\left(\frac{P_{\alpha/\gamma}}{P_{\alpha}}\right) \tag{2.6}$$

Using equation 2.3 we obtain,

$$\frac{\partial}{\partial w_{ij}} (P_{\alpha\beta/\gamma}) = \frac{\tau}{2} P_{\alpha\beta/\gamma} (S_i^{\alpha\beta/\gamma} S_j^{\alpha\beta/\gamma} - \sum_{\alpha'\beta'} P_{\alpha\beta/\gamma} S_i^{\alpha'\beta'/\gamma} S_j^{\alpha'\beta'/\gamma}) \tag{2.7}$$

Introducing the symbol $\langle x \rangle$ for the average value of x, then we can write the learning rule combining equation 2.5-2.7 as,

$$w_{ij}^{new} = w_{ij}^{old} + \frac{\tau\eta}{2} \cdot \sum_{\gamma} P_{\gamma} \sum_{\alpha,\beta} P_{\alpha\beta/\gamma} \log\left(\frac{P_{\alpha/\gamma}}{P_{\alpha}}\right) \cdot (S_i^{\alpha\beta/\gamma} S_j^{\alpha\beta/\gamma} - \langle S_i^{\alpha\beta/\gamma} S_j^{\alpha\beta/\gamma} \rangle) \tag{2.8}$$

The interpretation of the obtained unsupervised learning rule is interesting. A Hebbian term is given by the $S_i^{\alpha\beta/\gamma} S_j^{\alpha\beta/\gamma}$ in equation 2.8 and is the actual correlation between the neurons in the state $\alpha\beta$ given that γ is presented at the input. The second term $-\langle S_i^{\alpha\beta/\gamma} S_j^{\alpha\beta/\gamma} \rangle$ 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 a measure for the information transmitted from neuron to neuron in each possible state

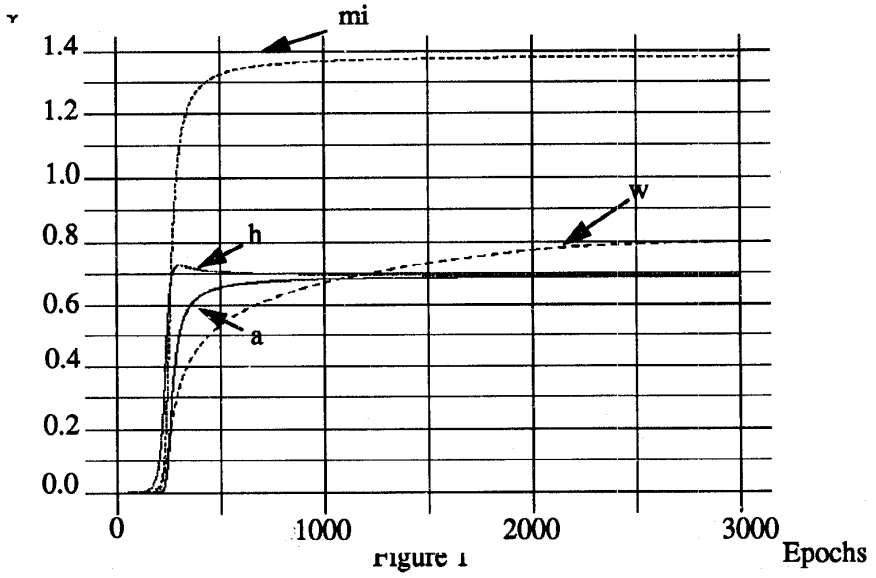

Figure 1

3.0 Results and Simulations

In order to see the convergence of the learning we simulate first the learning evolution for one neuron with one input. The two possible inputs 1 and -1 were presented with equal probability. The results for this very simple experiment are shown in figure 1. The maximization of the

mutual information (mi) until the maximal value (log 2) is observed. The Hebbian (h)and anti-Hebbian (a) terms are also plotted. Both terms increase so that they mutually cancel out when the number of epochs increases, so that this effect cause the convergence of the weight (w). The next test of our learning paradigm was performed by using the "encoder problem", which is the same problem solved by the supervised Boltzmann machine in Ackley et al. [1]. The input patterns for the encoder N-n consisting of N different patterns with N inputs where only one has the value 1 and the other the value -1. We assume the distribution of the patterns uniform ($P_\gamma = N^{-1}$). In all cases we have used $T = 0.1$ and $\eta = 0.01$. Three cases were studied: Encoder 4-2, 5-3 and 40-6. We didn't use hidden units. Figure 2 shows the evolution of the mutual information until the maximal values log(4) and log(5) are reached for the encoder problem 4-2 and 5-3 respectively. Is interesting to remark that for the three cases (4-2, 5-3 and 40-6) perfect binary data compression were obtained after training , that means for each different input pattern a different code with probability one is obtained.When redundant number of neurons are used the mutual information try to decorrelate the neurons so that distributed coding is obtained. This effect is also shown by Ackley et al [1] for the supervised Boltzmann machine.

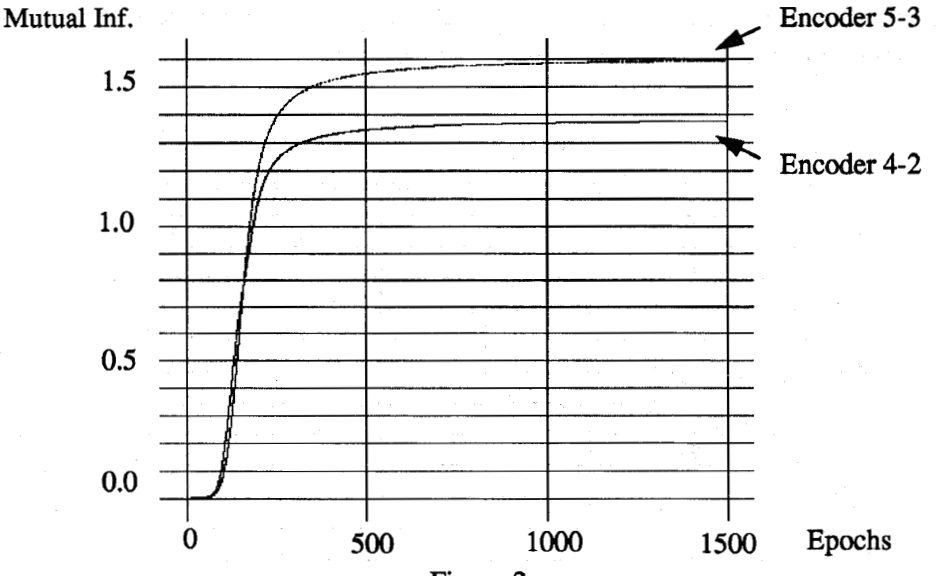

Figure 2

References:

- [1] Ackley D., Hinton G. and Sejnowski, 1985,"A Learning Algorithm for Boltzmann Machines", Cognitive Science, **9**, 147-169.
- [2] Linsker R., 1988, "Self-organization in a perceptual network", Computer, **21**, 105.
- [3] Linsker R., 1989, "How to generate ordered maps by maximizing the mutual information between input and output signals", Neural Computation, **1**, 402-411.
- [4] Linsker R., 1992, "Local Synaptic Learning Rules Suffice to Maximize Mutual Information in a Linear Network", Neural Computation, **4**, 691-702.
- [5] Shannon C., 1948, "A mathematical theory of communication", Bell System Technical Journal, **7**, 379-423.