Multilinear perceptron convergence theorem

H.-O. Carmesin*
Institut für Theoretische Physik, Universität Bremen, 28334 Bremen, Germany
(Received 22 February 1994)

An adaptive perceptron with multilinear couplings is introduced. While an adaptive perceptron exhibits severe shortcomings if it is applied to complex tasks, this is not so for the adaptive multilinear perceptron.

PACS number(s): 87.10.+e, 89.80.+h, 89.70.+c

INTRODUCTION

The perceptron is a traditional and important neural network model [1,2]. In its simplest version it has an input layer and an output layer. However, it cannot perform any desired task with these two layers. In more advanced versions it can perform any desired task. However, there is no learning algorithm known that converges for each desired task to a configuration of couplings so that the perceptron can perform the desired task. Thus in any case the perceptron exhibits essential shortcomings [2]. Here these shortcomings are completely overcome by a multilinear [3] perceptron, which is a perceptron that has couplings connecting pairs of neurons, triples of neurons, quadruples of neurons, and so forth.

DEFINITION OF THE MULTILINEAR PERCEPTRON

The multilinear perceptron consists of N input neurons $s_i=\pm 1$, $i=1,\cdots,N$, and N output neurons $\bar{s}_i=\pm 1$, $i=1,\cdots,N$. In addition, the network has one neuron s_0 that takes the value 1 at any time in any case. It has bilinear couplings J_{ij}, trilinear couplings J_{ijk}, and so forth. All couplings act from the input layer to the output layer. Because the network has only an input layer and an output layer, the neuronal dynamics can be expressed as an input-output mapping as follows. The state of an output neuron $\bar{s}_i$ is determined by the states of the input neurons s_i and of the couplings

$$\bar{s}_i=\mathrm{sgn}\left[J_i s_0+\sum_{j=1}^{N} J_{ij}s_j+\sum_{j<k=1}^{N} J_{ijk}s_j s_k + \sum_{j<k<l=1}^{N} J_{ijkl}s_j s_k s_l + \cdots + J_{i1,2\ldots N}s_1 s_2\ldots s_N\right] . \quad (1)$$

Here sgn denotes the signum function.

*FAX: 0421 218 4869.

DEFINITION OF A TASK

A task is an input-output mapping. That is, to each configuration $\{\xi_i^\mu\}$ of states of input neurons one configuration $\{\bar{\xi}_i^\mu\}$ of output neutrons is desired. Because there are 2^N configurations of input neurons, the index μ takes the values $\mu=1,\ldots,2^N$. It is convenient to denote the mapping from one input configuration $\{\xi_i^\mu\}$ to one output configuration $\{\bar{\xi}_i^\mu\}$ as an elementary task; a task consists of 2^N elementary tasks. By convention, the network performs a task when it performs correctly all 2^N elementary tasks.

MULTILINEAR PERCEPTRON EXISTENCE THEOREM

For each task there exists a multilinear perceptron that performs it. Proof: A desired state $\bar{\xi}_i^\mu$ is a function of the input states

$$\bar{\xi}_i^\mu=\bar{\xi}_i^\mu(\xi_1^\mu,\xi_2^\mu,\ldots,\xi_N^\mu)=\pm 1 . \quad (2)$$

This function can be expanded in terms of a Taylor series. A state is $+1$ or -1, thus $(\xi_i^\mu)^2=1$. Consequently, the expansion contains a finite number of terms, is exact, and looks as follows:

$$\bar{\xi}_i^\mu=a_{i0}+a_{i1}\xi_1^\mu+a_{i2}\xi_2^\mu+\cdots + a_{iN}\xi_N^\mu+a_{i12}\xi_1^\mu\xi_2^\mu+\cdots+a_{i12\ldots N}\xi_1^\mu\xi_2^\mu\ldots\xi_N^\mu . \quad (3)$$

The following couplings are constructed:

$$J^*_{ik\ldots l}=a_{ik\ldots l} . \quad (4)$$

By inserting Eq. (4) into Eq. (1) and ξ_i^μ for s_i, one obtains

$$\bar{s}_i=\mathrm{sgn}(a_{i0}s_0+a_{i1}\xi_1^\mu+\cdots+a_{i12\ldots N}\xi_1^\mu\xi_2^\mu\ldots\xi_N^\mu) . \quad (5)$$

The expansion is exact, thus the argument for the above signum function is $\bar{\xi}_i^\mu$ [see Eq. (3)]. Consequently, $\bar{s}_i=\bar{\xi}_i^\mu$. Thus the network performs the task.

DEFINITION OF THE LEARNING ALGORITHM

Next the usual [1,4] perceptron learning algorithm is generalized. In order to express the learning algorithm in a coherent and simple manner, one may use a vector notation as follows:

$$\zeta^\mu=(s_0,\xi^\mu_1,\xi^\mu_2,\xi^\mu_3,\ldots,\xi^\mu_N,\xi^\mu_1\xi^\mu_2,\xi^\mu_1\xi^\mu_3,\ldots,\xi^\mu_1\xi^\mu_N,\xi^\mu_1\xi^\mu_2\xi^\mu_3,\xi^\mu_1\xi^\mu_2\xi^\mu_4,\ldots,\xi^\mu_1\xi^\mu_2,\xi^\mu_N)\ . \tag{6}$$

An analogous vector $\mathbf{J}_i$ is formed for each i:

$$\mathbf{J}_i=(J_i,J_{i1},J_{i2},\ldots,J_{i1N},J_{i12},J_{i13},\ldots,J_{i1N},\ldots,J_{i12\ldots N})\ . \tag{7}$$

The scalar product may be denoted by $\langle\cdots|\cdots\rangle$, it is the sum over products of corresponding components. For instance, for the nth elementary task the input neurons take the states that occur in ζ^μ, and $\bar{s}_i$ is obtained from Eq. (1) as follows:

$$\bar{s}_i=\mathrm{sgn}(\langle\mathbf{J}_i|\zeta^\mu\rangle)\ . \tag{8}$$

Roughly speaking, the couplings are updated if the network did not process an elementary task as desired. This is done in a manner that is reminiscent of a generalized [3] Hopfield rule and of multilinear spin glasses [5]:

$$\mathbf{J}_i\to\mathbf{J}_i+\Theta(-\bar{s}_i\bar{\xi}^\mu_i)\bar{\xi}^\mu_i\zeta^\mu\ . \tag{9}$$

This prescription is made precisely in terms of a learning algorithm. The *multilinear perceptron learning algorithm* is defined as follows: One may start with any coupling state $(\mathbf{J}_1,\mathbf{J}_2,\ldots,\mathbf{J}_N)$, with the squared norm $\langle\mathbf{J}_i|\mathbf{J}_i\rangle=\frac{1}{4}$ for all $i=1,\ldots,N$. Next, one may normalize $\mathbf{J}^*_i$ and $2\zeta^\mu$, i.e., $\langle\mathbf{J}^*_i|\mathbf{J}^*_i\rangle=1$, and $\langle\zeta^\mu|\zeta^\mu\rangle=\frac{1}{4}$. That is, appropriate prefactors are assumed in Eqs. (4), (6), and (7): Sweep for $\mu:=1$ to 2^N; do update if $\bar{\xi}^\mu_i\langle\mathbf{J}_i|\zeta^\mu\rangle<0$, then $\mathbf{J}_i:=\mathbf{J}_i+\bar{\xi}^\mu_i\zeta^\mu$; if $(\mathbf{J}_1,\ldots,\mathbf{J}_N)$ has been changed during sweep, goto sweep; stop.

MULTILINEAR PERCEPTRON CONVERGENCE THEOREM

For each task, the multilinear perceptron learning algorithm stops after a finite number of steps. The proof is a generalization of a proof [4] of the perceptron convergence theorem. Proof: One considers any task. For the couplings constructed for the task in, Eq. (4) one obtains

$$\tfrac{1}{2}\geq\bar{\xi}^\mu_i\langle\mathbf{J}^*_i|\zeta^\mu\rangle>\delta\geq 0 \quad \text{for each } i=1,\ldots,N \text{ and some } \delta\ . \tag{10}$$

Because the network with the couplings performs the task, so $\bar{s}_i=\bar{\xi}^\mu_i$ for the nth elementary task, thus Eq. (8) implies Eq. (10). Let $\mathbf{J}^{(n)}_i$ denote the vector $\mathbf{J}_i$ after n updates have been performed. So n times we had $\langle\mathbf{J}_i|\zeta^\mu\bar{\xi}^\mu_i\rangle\leq 0$ (for all $i=1,\ldots,N$) for some ζ^μ and added ζ^μ, so that $\mathbf{J}_i\to\mathbf{J}_i+\zeta^\mu\bar{\xi}^\mu_i$. By the Cauchy-Schwartz inequality,

$$\frac{\langle\mathbf{J}^*_i|\mathbf{J}^{(n)}_i\rangle}{|\mathbf{J}^{(n)}_i|}\leq 1 \quad \text{for all } i=1,\ldots,N\ , \tag{11}$$

because $\mathbf{J}^*_i$ is renormalized. Equation (11) leads to an upper bound for n. To slow this, we estimate the numerator and the denominator in Eq. (11) separately. For the numerator, since $\langle\mathbf{J}^*_i|\zeta^\mu\bar{\xi}^\mu_i\rangle>\delta$, we have

$$\begin{aligned}\langle\mathbf{J}^*_i|\mathbf{J}^{(n)}_i\rangle&=\langle\mathbf{J}^*_i|(\mathbf{J}^{(n-1)}_i+\zeta^\mu\bar{\xi}^\mu_i)\rangle\\&>\langle\mathbf{J}^*_i|\mathbf{J}^{(n-1)}_i\rangle+\delta\\&>\cdots>\langle\mathbf{J}^*_i|\mathbf{J}^{(0)}_i\rangle+n\delta=a+n\delta\ ,\end{aligned} \tag{12}$$

where $-\frac{1}{2}\leq a\leq\frac{1}{2}$. The denominator is estimated by

$$\begin{aligned}|\mathbf{J}^{(n)}_i|^2&=\langle\mathbf{J}^{(n)}_i|\mathbf{J}^{(n)}_i\rangle\\&=\langle(\mathbf{J}^{(n-1)}_i+\zeta^\mu\bar{\xi}^\mu_i)|(\mathbf{J}^{(n-1)}_i+\zeta^\mu\bar{\xi}^\mu_i)\rangle\\&=|\mathbf{J}^{(n-1)}_i|^2+2\langle\mathbf{J}^{(n-1)}_i|\zeta^\mu\bar{\xi}^\mu_i\rangle+|\zeta^\mu\bar{\xi}^\mu_i|^2\leq|\mathbf{J}^{(n-1)}_i|^2+1\leq\cdots\leq n+1\ .\end{aligned} \tag{13}$$

Because $\langle\mathbf{J}^{(n-1)}_i|\zeta^\mu\bar{\xi}^\mu_i\rangle\leq 0$; otherwise $\zeta^\mu\bar{\xi}^\mu_i$ would not have been added. By inserting Eqs. (12) and (13) into Eq. (11), one obtains

$$\frac{a+n\delta}{\sqrt{n+1}}\leq\frac{\langle\mathbf{J}^*_i|\mathbf{J}^{(n)}_i\rangle}{|\mathbf{J}^{(n)}_i|}\leq 1\ . \tag{14}$$

Thus a and δ are the relevant parameters for the convergence of the process; due to the normalizations one obtains $\delta\leq\frac{1}{2}$ and $|a|\leq 1/2$ [see Eqs. (10) and (12)]. Thus the above inequality has solutions for n, and the process has at most $n_{\max}$ iterations with

$$n_{\max}\leq\frac{1}{2\delta^2}\left[1-2a\delta+\sqrt{(1-2a\delta)^2+4\delta^2(1-a^2)}\right]\xrightarrow{\delta\to 0}\delta^{-2}\ . \tag{15}$$

POSSIBLE APPLICATIONS

The typical property of the network is not to compress information, because one uses a network with 2^N weights, each one of which has a solution of N bits, to encode 2^N bits of information. In contrast, the typical property of the network is "precision" and "the ability to adapt to novel situations." This is illustrated with an example. If one starts with an incomplete set of elementary tasks and trains these, then the network performs these correctly. Later, novel elementary tasks may be introduced and trained while the network is processing. Thereby the current coupling state plays the role of the initial coupling state in the convergence theorem, so the theorem can be applied; thus the network adapts precisely to the new extended set of elementary tasks.

For the field of biology, the network can be interpreted as a model for neuropils (sets of multiply connected neurons in the brain), because the presynaptic neurons can act in any functional manner on a postsynaptic neuron [6].

Finally, there are many situations in extrapolation, learning, memory, and perception that cannot be expressed in terms of tasks and elementary tasks, due to ambiguous, incomplete, or preliminary information [6,7]. While bilinear networks do not even converge in situations with complete information, multilinear networks do, and are therefore good candidates for adaptation and performance with incomplete information.

CONCLUSION

The perceptron has been generalized to the multilinear perceptron. The perceptron learing algorithm has been generalized to the multilinear perceptron learning algorithm. It has been proven that the latter converges for any task. That is, the restrictions to the applicability of the perceptron to relatively simple tasks have been completely resolved. This result applies also to feedback networks [8].

ACKNOWLEDGMENTS

I acknowledge fruitful discussions with Martin Kreyscher, Florian Sander, and Helmut Schwegler.

[1] F. Rosenstslatt, *Principles of Neurodynamics* (Spartau, Washington, DC, 1961).

[2] M. Minsky and S. Papert, *Perceptron* (MIT, Cambridge, MA, 1969).

[3] H.-O. Carmesin, Phys. Lett. A **156**, 183 (1991).

[4] J. L. van Hemmen and R. Kühn, in *Models of Neural Networks,* edited by E. Domany, J. L. van Hemmen, and K. Schulten (Springer, Berlin, 1991).

[5] H.-O. Carmesin, Z. Phys. B **73**, 381 (1988).

[6] H.-O. Carmesin, *Theorie der Adaption* (Köster, Berlin, 1994).

[7] C. Basar-Eroglu, D. Strüber, M. Stadler, P. Kruse, and E. Basar, Int. J. Neurosci. **73**, 139 (1993).

[8] F. J. Pineda, Phys. Rev. Lett. **59**, 2229 (1987).