A polynomial training algorithm for calculating perceptrons of optimal stability

1995 J. Phys. A: Math. Gen. 28 2173

(http://iopscience.iop.org/0305-4470/28/8/011)

J. Phys. A: Math. Gen. 28 (1995) 2173–2181. Printed in the UK

A polynomial training algorithm for calculating perceptrons of optimal stability

Jorg Imhoff†

Universität Heidelberg, Institut für Theoretische Physik, Philosophenweg 19, D-69120 Heidelberg, Germany

Received 30 September 1994, in final form 23 December 1994

Abstract. RECOMI (Repeated correlation matrix inversion) is a polynomially fast algorithm for searching optimally stable solutions of the perceptron learning problem. For random unbiased and biased patterns it is shown that the algorithm is able to find optimal solutions, if any exist, in at worst $\mathcal{O}(N^4)$ floating point operations. Even beyond the critical storage capacity α_c the algorithm is able to find locally stable solutions (with negative stability) at the same speed. There are no divergent timescales in the learning process. A full proof of convergence cannot yet be given, only major constituents of a proof are shown.

Spin-glass models of neural networks and their application as an associative memory have been of great interest in the last few years [1–10]. One major issue of the field is the question of training networks, that is the construction of a synaptic matrix in order to store given information. In this paper I will present a training algorithm that is able to find solutions of the perceptron problem of optimal stability in finite time. Unlike other algorithms, such as Minover presented by Krauth and Mézard [5] or AdaTron by Anlauf and Biehl [6], this algorithm not only approximates optimal solutions, it actually finds the optimal solutions. Furthermore, there are no divergent timescales in the solution of the problem. Minover and AdaTron both have diverging training times as the critical storage capacity α_c is approached [6, 7], whereas this algorithm does not. Therefore it can also be used beyond α_c in the region of broken replica symmetry, where it finds local optima of negative stability. A similar algorithm was proposed by Ruján [8], which also finds optimal perceptrons in finite time, but cannot advance beyond α_c.

Like the pseudo-inverse solution of the perceptron problem [9, 10] this algorithm uses inversion of pattern correlation matrices for searching (optimal) perceptron couplings. As matrix inversion has to be done repeatedly, the algorithm was called RECOMI—Repeated correlation matrix inversion. As was shown by Opper [7] the problem of finding an optimal perceptron is the problem of finding the subset of embedded training patterns with minimal local fields. RECOMI is able to find this subset of patterns iteratively in finite time. The coupling vector is then just the pseudo-inverse of the respective pattern correlation matrix.

I consider a network of $N+1$ neurons $S_i = \pm 1$, $i = 1, \ldots, N+1$, coupled through synaptic efficacies J_{ij} (without taking self-couplings into account, i.e. $J_{ii} = 0 \quad \forall i$). The dynamics of the system is taken to be a simple zero-temperature Monte Carlo process:

$$S_i(t+1) = \mathrm{sgn}\left(\sum_{j(\neq i)} J_{ij} S_j(t) \right). \qquad (1)$$

† E-mail address: imhoff@hybrid.tphys.uni-heidelberg.de

The purpose of perceptron training algorithms is to find couplings J_{ij} such that p patterns $\underline{\eta}^\mu = (\eta_1^\mu, \ldots, \eta_{N+1}^\mu)^T$, $\eta_i^\mu = \pm 1$, $\mu = 1, \ldots, p$, become fixed points of the dynamics. That is

$$\eta_i^\mu \sum_{j(\neq i)} J_{ij} \eta_j^\mu \geqslant \kappa > 0 \qquad i = 1, \ldots, N+1 \qquad \mu = 1, \ldots, p\,. \tag{2}$$

The problem can be reformulated by looking at the single neurons (or simple perceptrons) of the network, e.g. neuron $N+1$. With

$$\xi_i^\mu \overset{\mathrm{def}}{=} \eta_{N+1}^\mu \eta_i^\mu \qquad i = 1, \ldots, N \qquad \mu = 1, \ldots, p \tag{3}$$

one now has to find couplings J_i, $i = 1, \ldots, N$, such that

$$h_\mu = \sum_i J_i \xi_i^\mu \geqslant \kappa > 0 \qquad \mu = 1, \ldots, p\,. \tag{4}$$

If the norm of $\underline{J}$ is fixed, e.g. $|\underline{J}| = 1$, it is possible to define what is meant by 'optimal solutions' of the given problem:

$$\text{maximize } \kappa = \min_\mu \{h_\mu\} \text{ under the constraint } \left|\underline{J}\right| = 1\,. \tag{5}$$

With maximal κ one expects to have maximum stability against input noise, i.e. maximal basins of attraction in a network of neurons.

From the point of view of mathematical optimization it suitable to reformulate the problem. With $\underline{J} \longrightarrow \underline{J}/|\kappa|$ one gets an equivalent formulation of problem (5):

$$\text{minimize } \left|\underline{J}\right| \text{ under the constraints } h_\mu = \underline{J}^T \underline{\xi}^\mu \geqslant +1\ \forall \mu \text{ (for } \kappa > 0) \tag{6}$$

$$\text{maximize } \left|\underline{J}\right| \text{ under the constraints } h_\mu = \underline{J}^T \underline{\xi}^\mu \geqslant -1\ \forall \mu \text{ (for } \kappa < 0)\,. \tag{7}$$

I will use this formulation of the problem later in this paper. Applying the Kuhn–Tucker theorem of optimization theory [11] it can be shown [7] (see also [6]) that an optimal solution, for $\kappa > 0$, can always be written in the form

$$\underline{J} = \sum_{\mu \in \Gamma} x_\mu \underline{\xi}^\mu \qquad \text{where} \quad x_\mu \geqslant 0 \quad \forall \mu \in \Gamma \tag{8}$$

with

$$h_\mu = \underline{J}^T \underline{\xi}^\mu \begin{cases} = \kappa & \mu \in \Gamma \\ > \kappa & \text{otherwise}\,. \end{cases} \tag{9}$$

For $\kappa < 0$ the same argument holds for all local optima, but with $x_\mu \leqslant 0$, $\forall \mu \in \Gamma$. Γ is the set of 'embedded' patterns, $\Gamma \subseteq \{1, \ldots, p\}$. The x_μ are called the embedding strengths of solution $\underline{J}$. Anlauf and Biehl have also shown [6] that for $\kappa > 0$ this solution is unique (which, in general, is not the case for $\kappa < 0$), i.e. two solutions $\underline{J}$ and $\underline{J}^*$ of the form (8) or (9) are always identical $\underline{J} \equiv \underline{J}^*$. Note that if $\{\xi_\mu \,|\, \mu \in \Gamma\}$ is a set of linearly independent vectors, e.g. if the patterns are in a general position and card$(\Gamma) \leqslant N$ the choice of the x_μ is unambiguous. On the other hand, if one has a solution of the form (8) or (9) it must be the global optimum of the problem.

In the following sections I am going to describe the RECOMI algorithm. RECOMI can solve the stated problem of finding optimal perceptrons of the form (8) or (9) in finite time, if the training patterns are in a general position, i.e. if every subset $\{\xi_\mu\}$ with not more than N elements (card$(\{\xi_\mu\}) \leqslant N$) is linearly independent. It does so in not more than $\mathcal{O}(N^4)$ floating point operations. There is no divergence of learning times at the critical storage capacity $\alpha_{\mathrm{c}} = 2$ (for unbiased random patterns), where $\alpha = p/N$. I am going to show this numerically. In the last section I will deduce some important constituents of a proof of convergence, unfortunately a full proof cannot yet be given. I will analyse the properties of locally stable solutions of the optimization problems (6) and (7). It can be shown that RECOMI always stops in a local optimum. If an optimal solution with $\kappa > 0$ exists, RECOMI must stop there. Otherwise it is going to stop in one of the locally stable solutions with $\kappa < 0$.

Description of the algorithm

RECOMI is an iterative algorithm. It calculates coupling vectors $\underline{J}^{(t)} = \sum_\mu x_\mu^{(t)} \underline{\xi}^\mu$ and finds after a finite number of iterations a solution of the form (8) and (9), if it exists. As we will see later, the algorithm must be initialized with positive embedding strengths $x_\mu^{(0)} \geqslant 0$, e.g. Hebbian couplings

$$\underline{J}^{(0)} = \sum_\mu \underline{\xi}^\mu .$$

For numerical stability $\underline{J}^{(t)}$ is normalized to 1 after each iteration. Let C_Γ be the correlation matrix of the patterns in $\Gamma \subseteq \{1, \dots, p\}$:

$$C_\Gamma = \left(\underline{\xi}^{\mu T} \underline{\xi}^\nu\right)_{\mu,\nu\in\Gamma} . \tag{10}$$

Iteration loop

Let $\underline{J}^{(t)}$ be given (from now on I drop the index t):

$$\underline{J} = \sum_{\mu=1}^{p} x_\mu \underline{\xi}^\mu \qquad (|\underline{J}| = 1) \tag{11}$$

$$\kappa = \min_\mu \{h_\mu\} = \min_\mu \{\underline{J}^T \underline{\xi}^\mu\} . \tag{12}$$

Let Γ be the subset of patterns with minimal local field h_μ:

$$\Gamma = \{\mu | h_\mu = \kappa\} . \tag{13}$$

We now want to alter $\underline{J}$

$$\underline{J} \quad \longrightarrow \quad \underline{J}' = \sum_{\mu=1}^{p} \left(x_\mu + \varepsilon \Delta x_\mu\right) \underline{\xi}^\mu \tag{14}$$

so that for all patterns in Γ the local fields grow equally

$$h'_\mu = \underline{J}'^T \underline{\xi}^\mu = \kappa + \varepsilon \qquad \forall \mu \in \Gamma . \tag{15}$$

We therefore choose $\Delta \underline{x}$ to be the pseudo-inverse [9, 10] of the patterns in Γ:

$$\Delta x_\mu = \begin{cases} \sum_{\nu\in\Gamma} (C_\Gamma^{-1})_{\mu\nu} & \mu \in \Gamma \\ 0 & \text{otherwise} . \end{cases} \tag{16}$$

If the training patterns $\underline{\xi}_\mu$ are in general position, C_Γ becomes singular if and only if the number of patterns in Γ, card(Γ), is greater than N. Then RECOMI must stop, with $\underline{J}^{(t)}$ being the best solution found. Nevertheless RECOMI is able to find optimal solutions as I will show in the last section of this paper.

Now we want to determine the learning rate ε in a way that *all* local fields h'_μ are greater or equal $\kappa + \varepsilon$:

$$h'_\mu = \underline{J}'^T \underline{\xi}^\mu \geqslant \kappa + \varepsilon \qquad \forall \mu \in \{1, \dots, p\} \tag{17}$$

where $\varepsilon = \varepsilon_\mu$ is the value of the learning rate with which we get the equality $h'_\mu = \kappa + \varepsilon$ for pattern μ:

$$\varepsilon_\mu = \frac{h_\mu - \kappa}{1 - \sum_{\nu\in\Gamma} C_{\mu\nu}\Delta x_\nu} . \tag{18}$$

To fulfil equation (17) ε must be smaller or equal to all relevant, i.e. all positive, ε_μ. We therefore define the set Φ:

$$\Phi = \{\varepsilon_\mu | \mu \notin \Gamma \quad \text{and} \quad 0 < \varepsilon_\mu < \infty\} . \tag{19}$$

If Φ is not empty we can determine ε as

$$\varepsilon = \min \Phi . \tag{20}$$

If Φ is empty, we set $\varepsilon = \infty$, i.e. $\underline{J}' = \sum_{\mu\in\Gamma} \Delta x_\mu \underline{\xi}^\mu$, and stop the iteration.

Now $\underline{J}^{(t+1)} = \underline{J}'/|\underline{J}'|$ and we continue at the beginning of the iteration loop. It is easy to show that always $\kappa^{(t+1)} = (\kappa^{(t)} + \varepsilon)/|\underline{J}'| > \kappa^{(t)}$ (see the appendix). If no solution with positive κ can be found the algorithm typically stops with $\underline{J}' = \underline{0}$, as will be shown later. (It should be noted that this is the most sensitive part of the algorithm. Rounding errors must be controlled when calculating the norm of $\underline{J}'$.) Then $\underline{J}^{(t)}$ is taken as the best solution found by RECOMI.

Optimal RECOMI

The algorithm I have described so far does not yet find optimal solutions of the form (8) and (9). As the changes of embedding strengths Δx_μ might be negative in (16) the x_μ might also become negative in the end. But already this version of the algorithm does find nearly optimal solutions $\kappa > 0$, as can be seen in figure 1, where I compare results for unbiased random patterns ($N = 100$) with Gardner's result [3]. Therefore I refer to this version of RECOMI as 'nearly optimal RECOMI'.

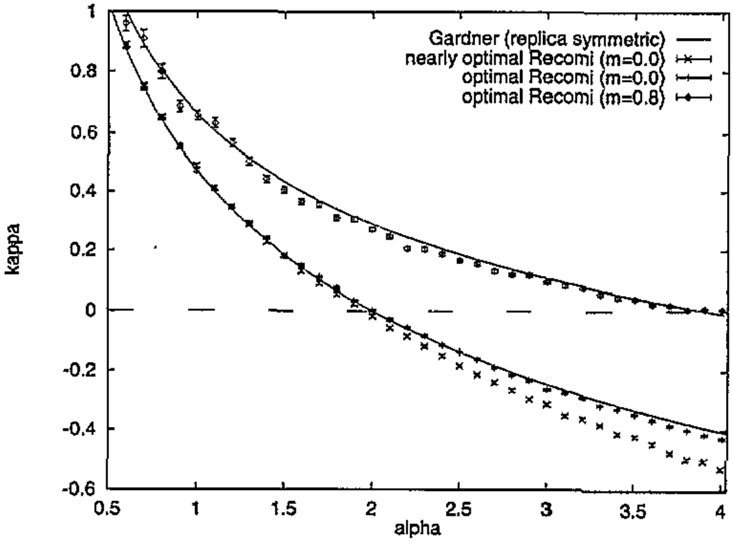

Figure 1. Comparison of RECOMI with Gardner's result, $N = 100$, 100 sets of unbiased ($m = 0$) and biased ($m = 0.8$) random binary patterns for each measurement.

To find optimal solutions of the form (8) and (9) it is necessary to start with positive embedding strengths $x_\mu \geqslant 0$, and to make sure that they stay positive throughout the iteration, i.e. $\Delta x_\mu \geqslant 0$. This is possible by altering (16). Γ must be replaced by a subset $\Gamma' \subseteq \Gamma$ with the following properties:

$$\Gamma' \subseteq \Gamma \tag{21}$$

$$\Delta x_\mu = \sum_{\nu \in \Gamma'} \left(C_{\Gamma'}^{-1}\right)_{\mu\nu} \geqslant 0 \qquad \forall \mu \in \Gamma' \tag{22}$$

$$\left(\sum_{\nu \in \Gamma'} \Delta x_\nu \underline{\xi}^\nu\right)^T \underline{\xi}^\mu \geqslant 1 \qquad \forall \mu \in \Gamma \,. \tag{23}$$

It is always possible to find such a subset Γ' (as long as C_Γ itself is regular), because $\sum_{\nu \in \Gamma'} \Delta x_\nu \underline{\xi}^\nu$ then is the (unique) optimal perceptron for the correct mapping of the patterns $\mu \in \Gamma$.

Γ' can easily be determined. The following algorithm proved to work in all cases tested (about $\mathcal{O}(10^5)$ algorithm runs). I cannot yet prove its convergence analytically. This has to be done in later work. To find Γ' one can proceed as follows:

(i) Start with $\Gamma' = \Gamma$.

(ii) Calculate $\Delta x_\mu = \sum_{\nu \in \Gamma'} \left(C_{\Gamma'}^{-1}\right)_{\mu\nu} (\mu \in \Gamma')$; $\Delta x_\varrho = \min_\mu\{\Delta x_\mu\}$; if $\Delta x_\varrho < 0$ remove ϱ from Γ' and go to (ii) else go to (iii).

(iii) Calculate $\Delta h_\mu = (\sum_{\nu \in \Gamma'} \Delta x_\nu \underline{\xi}^\nu)^T \underline{\xi}^\mu$ $(\mu \in \Gamma \setminus \Gamma')$; $\Delta h_\sigma = \min_\mu\{\Delta h_\mu\}$; if $\Delta h_\sigma < 1$ add σ to Γ' and go to (ii) else STOP.

By replacing Γ by Γ' in (16) RECOMI is able to find optimal solutions. I refer to this improved version of the algorithm as 'optimal RECOMI'. In figure 2 I check for unbiased random binary patterns ($N = 100$), how often the algorithm stops in optimal solutions with $\kappa > 0$, and in locally optimal solutions with $\kappa < 0$. For every value of $\alpha = p/N$ 100 different pattern sets are tested. In very rare cases (not in this figure) the algorithm only gets close to but does not reach optimal solutions: trying to invert nearly singular correlation matrices can cause failure of the inversion subroutines.

In figure 1 I compare results for unbiased and biased random binary patterns with Gardner's result [3]. The patterns η_i^μ are chosen with a probability distribution $p(\eta_i^\mu) =$

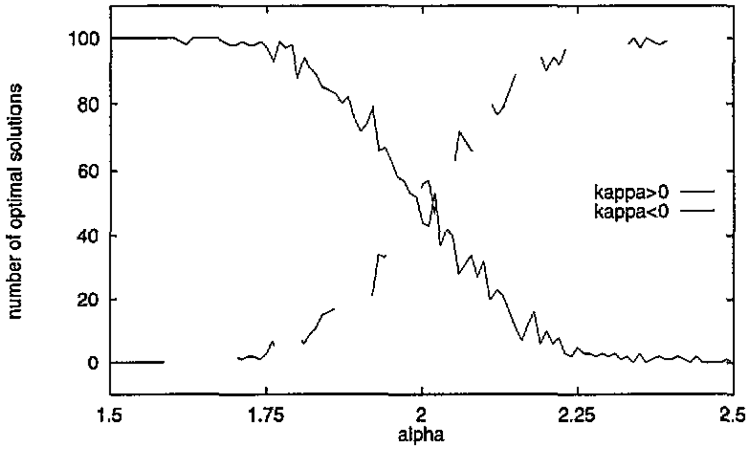

Figure 2. Optimal RECOMI, $N = 100$, 100 sets of unbiased random binary patterns for each value of α.

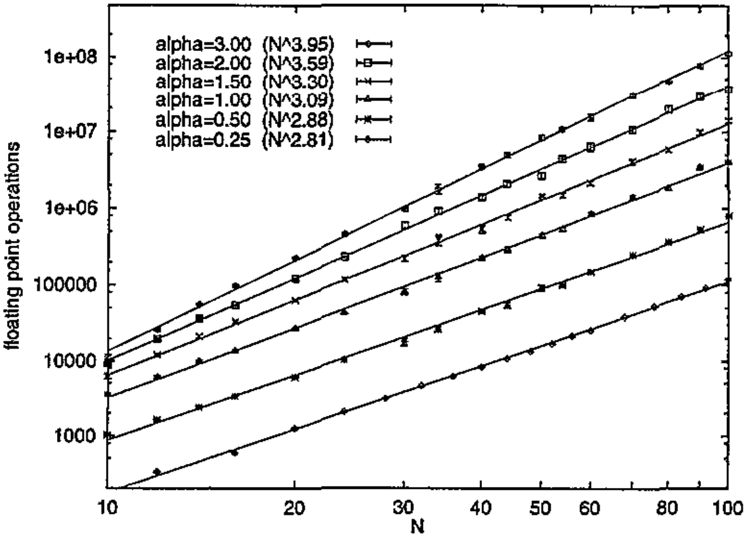

Figure 3. Convergence time against system size for optimal RECOMI (unbiased random binary patterns).

$\frac{1}{2}(1-m)\,\delta(\eta_i^\mu+1)+\frac{1}{2}(1+m)\,\delta(\eta_i^\mu-1)$, using $m=0$ (unbiased) and $m=0.8$ (biased), and the ξ_i^μ calculated according to (3). Within the error bounds there is no difference to be seen between optimal and nearly optimal solutions below α_c ($\kappa>0$). In the range of replica symmetry breaking $\alpha>\alpha_c$ ($\kappa<0$) optimal RECOMI clearly performs better than the simpler version of the algorithm. Here it cannot be expected that the algorithm finds a global stability optimum, as it gets trapped in one of the many local optima, which will be shown in the last section of this paper. Note that for the biased patterns ($m=0.8$) at $N=100$ one still has to take finite-size effects into account: the measured points are all optimal solutions, but yet still lie a little bit below the Gardner curve. Also note that the theoretical lines are all calculated in replica symmetric approximation, i.e. they must be corrected for negative κ, where replica symmetry is no longer valid.

In figure 3 I train perceptrons of different sizes N with unbiased random binary patterns. Convergence time is plotted against system size N for different values of the storage capacity α. The most expensive part of the algorithm, in the large-N limit, is matrix inversion, which is of $\mathcal{O}(N^3)$ for each single inversion. Nearly optimal RECOMI therefore is, in the worst case, of $\mathcal{O}(\sum_{i=1}^{N} i^3)=\mathcal{O}(N^4)$, as card($\Gamma$) grows at least by one in each iteration step. For optimal RECOMI one cannot give such a simple derivation of convergence times, as card(Γ) can also shrink in the learning process. But here convergence time is also bounded from above by $\mathcal{O}(N^4)$. In figure 3 I count the number of floating point operations (+−*/) optimal RECOMI needs to find solutions. As below $N=100$ convergence time is still dominated by other operations apart from matrix inversion, I only plot the matrix inversion part here. All other operations are of $\mathcal{O}(N^3)$ or below. Just as predicted for nearly optimal RECOMI the optimal version of the algorithm converges in $\mathcal{O}(N^4)$ or less floating point operations.

In figure 4 I plot convergence time (i.e. number of floating point operations) against the storage capacity α. Again the perceptron ($N=100$) was trained with unbiased random binary patterns. There is no divergence at $\alpha=\alpha_c=2$. For small α the two versions of the algorithm differ only slightly, as nearly optimal RECOMI also often finds optimal solutions (see also figure 1). For larger values of α the convergence times evolve differently.

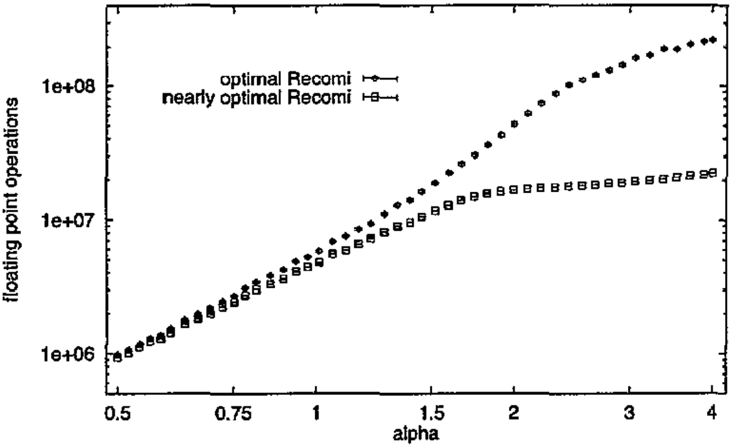

Figure 4. Convergence time against storage capacity α (unbiased random binary patterns, $N = 100$). There is no divergence at $\alpha_c = 2$.

Analysis of local stability optima: towards a proof of convergence

I cannot yet give a full proof of convergence of RECOMI, but some major components can already be deduced. For this reason I want to consider the role of local stability optima. It is useful here to use the problem formulations (6) (for $\kappa > 0$) and (7) (for $\kappa < 0$). If I write a ±-sign in the following text, the + always refers to the case $\kappa > 0$ and the − to $\kappa < 0$.

The problem (6) and (7) can now be formulated as

$$\text{minimize} f(\underline{x}) = \pm\, \underline{J}^T \underline{J} = \pm\, \underline{x}^T C \underline{x}$$

under the constraints:

$$h_\mu = \underline{J}^T \underline{\xi}^\mu = (C\underline{x})_\mu \geqslant \pm 1 \qquad \mu = 1, \ldots, p\,. \tag{24}$$

Γ is the set of patterns with minimal local field:

$$\Gamma = \left\{\, \mu \;\middle|\; h_\mu = (C\underline{x})_\mu = \pm 1 \,\right\}. \tag{25}$$

Let Ω be the set of all possible search directions $\Delta\underline{x}$, which do not violate the inequality constraints (24):

$$\Omega = \left\{ \Delta\underline{x} \in \mathbb{R}^p | (C\Delta\underline{x})_\mu \geqslant 0 \qquad \forall\mu \in \Gamma \right\}. \tag{26}$$

A solution $\underline{J}$ is locally optimal if and only if

$$[\nabla_x f(\underline{x})]^T \Delta\underline{x} = \pm 2\underline{x}^T C \Delta\underline{x} \geqslant 0 \qquad \forall \Delta\underline{x} \in \Omega\,. \tag{27}$$

I now prove the important theorem, that if there is a solution with positive stability $\kappa > 0$ there cannot be locally stable solutions $\underline{J}$ with negative stability $\kappa < 0$ and $x_\mu \geqslant 0$ $\forall\mu$, $\sum_\mu x_\mu > 0$.

If there is a solution with $\kappa > 0$ there must be a solution of the form (e.g. the optimal perceptron)

$$\underline{J}^* = \sum_\mu x^*_\mu \underline{\xi}^\mu \qquad \text{with} \qquad \underline{J}^{*T} \underline{\xi}^\mu = (C\underline{x}^*)_\mu \geqslant \kappa^* > 0 \qquad \forall\mu\,. \tag{28}$$

Let us assume $\underline{J} = \sum_\mu x_\mu \underline{\xi}^\mu$ is locally optimal with $\kappa = \min_\mu\{\underline{J}^T \underline{\xi}^\mu\} < 0$ and $x_\mu \geqslant 0\ \forall\mu$, $\sum_\mu x_\mu > 0$. That means (equation (27))

$$\underline{x}^T C \Delta\underline{x} \leqslant 0 \qquad \forall \Delta\underline{x} \in \Omega\,. \tag{29}$$

As $(C\underline{x}^*)_\mu \geqslant \kappa^* > 0\ \forall\mu$, we have

$$\Delta\underline{x} \overset{\text{def}}{=} \underline{x}^* \in \Omega \tag{30}$$

$$\underline{x}^T C\Delta\underline{x} = \underline{x}^T C\underline{x}^* = \sum_\mu x_\mu (C\underline{x}^*)_\mu \geqslant \kappa^* \sum_\mu x_\mu > 0 \tag{31}$$

in contradiction to (29)! Therefore such a vector $\underline{J}$ cannot exist. We will see below that optimal RECOMI always stops in (local) optima which, by definition of the algorithm, are of the form $x_\mu \geqslant 0\ \forall\mu$ and $\sum_\mu x_\mu > 0$. So if there is any solution with $\kappa > 0$ RECOMI can only stop in the global optimum of the problem, because then there are no other optima of that form.

To show this, I have to make several assumptions, which I cannot yet prove: (i) the algorithm described in section 'optimal RECOMI' for deriving Γ' really always works. (ii) The size of Γ, card(Γ), grows not more than by one in each iteration step, especially not from card(Γ) $< N$ to card(Γ) $> N$. (iii) RECOMI really terminates in finite time. About this last point one can only say that $\kappa^{(t)}$ is a strictly monotonical function of t (see the appendix), i.e. there is always an attractor of the training dynamics.

If these three assumptions are correct, RECOMI stops in a (local) optimum, which is the global one, if solutions $\kappa > 0$ exist. To show this I have to consider the three possible ways the algorithm does stop: (i) Φ is empty, i.e. ε becomes infinite, (ii) $\underline{J}'$ is zero and (iii) C_Γ is singular.

(i) Φ is empty: This is the most simple case. Then, by definition, $\underline{J}' = \sum_{\mu\in\Gamma'} \Delta x_\mu \underline{\xi}^\mu$, which is an optimal solution of the form (8) and (9). This is the usual way RECOMI stops if solutions $\kappa > 0$ exist.

(ii) $\underline{J}'$ is zero: then $\underline{J}^{(t)} = -\varepsilon \sum_{\mu\in\Gamma'} \Delta x_\mu \underline{\xi}^\mu$. Applying the Kuhn–Tucker theorem this is a locally stable solution for $\kappa < 0$ (just like (8) and (9) for $\kappa > 0$). As $\underline{J}^{(t)}$ is coded in the form $x_\mu \geqslant 0\ \forall\mu$ and $\sum_\mu x_\mu > 0$ there cannot be solutions with $\kappa > 0$ as was shown above. This is the usual way RECOMI stops if no solutions $\kappa > 0$ exist.

(iiii) C_Γ is singular: then card(Γ) $> N$ (because the training patterns are in a general position). According to our assumption, card(Γ) must have been N in the iteration step before. Γ' must have been equal to Γ because otherwise card(Γ) would not have grown. As $\{\xi^\mu | \mu \in \Gamma\}$ does span $\mathbb{R}^N$, $\underline{J}^{(t-1)}$ is completely determined by the local fields $\underline{J}^{(t-1)T}\underline{\xi}^\mu$ $\mu \in \Gamma$, i.e. $\underline{J}^{(t-1)} \sim \sum_{\mu\in\Gamma} \Delta x_\mu \underline{\xi}^\mu$, which is a local optimum. Therefore case (iii), in principal, never occurs, the algorithm stops before in (i) or (ii).

In practice case (iii) does occur, as sometimes nearly singular correlation matrices cannot be inverted by the inversion subroutines because of numerical restrictions.

Conclusion

In this paper I have presented a perceptron learning algorithm, which is able to find the optimal perceptron in finite time, i.e. in $\mathcal{O}(N^4)$ floating point operations. The algorithm even works beyond the critical storage capacity α_c, where it finds solutions of negative stability that are locally optimal. Calculating the stability curve $\kappa(\alpha)$ for random training patterns exactly reproduces Gardner's predictions [3]. A full proof of convergence could not yet be given, but major constituents were already shown. As the algorithm works very reliably, it can be expected that a full proof of convergence can be found. Furthermore, it is planned to generalize the algorithm to two-layer perceptrons with fixed output. First results are very promising, yet it cannot be expected that the algorithm finds globally optimal solutions, because replica-symmetry breaking effects are very strong in this case.

Acknowledgments

I would like to thank Professor H Horner, Dr R Kühn and Professor H G Bock for valuable discussions. This work was sponsored by the IWR (Interdisciplinary Center for Scientific Computing), Universität Heidelberg.

Appendix

In this appendix I will show that $\kappa^{(t)}$ is a strictly monotonic function of t:

$$\kappa^{(t+1)} = \frac{\kappa^{(t)} + \varepsilon}{|\underline{J}'|} \tag{A1}$$

$$\varrho \overset{\text{def}}{=} \left(\sum_{\mu} \Delta x_\mu \underline{\xi}^\mu \right)^2 = \sum_{\mu\nu} \Delta x_\mu (C_\Gamma)_{\mu\nu} \Delta x_\nu = \sum_{\mu\nu\lambda} \Delta x_\mu (C_\Gamma)_{\mu\nu} (C_\Gamma^{-1})_{\nu\lambda} = \sum_{\mu} \Delta x_\mu \geqslant 0 \tag{A2}$$

$$\underline{J}'^T \underline{J}' = \left(\underline{J}^{(t)} + \varepsilon \sum_{\mu\in\Gamma} \Delta x_\mu \underline{\xi}^\mu \right)^2 = 1 + 2\varepsilon\kappa^{(t)}\varrho + \varepsilon^2\varrho \geqslant 0 \qquad \forall \varepsilon \in \mathbb{R} \tag{A3}$$

$$\text{e.g.} \quad \varepsilon = -\kappa^{(t)} \quad \Longrightarrow \quad 1 - \kappa^{(t)^2}\varrho \geqslant 0 \tag{A4}$$

$$\frac{\mathrm{d}}{\mathrm{d}\varepsilon}\kappa^{(t+1)} = \left(1 + 2\varepsilon\kappa^{(t)}\varrho + \varepsilon^2\varrho\right)^{-3/2}\left(1 - \kappa^{(t)^2}\varrho\right) \geqslant 0 \tag{A5}$$

$\frac{\mathrm{d}}{\mathrm{d}\varepsilon}\kappa^{(t+1)} = 0$ if and only if RECOMI stops in a (local) optimum:

$$\frac{\mathrm{d}}{\mathrm{d}\varepsilon}\kappa^{(t+1)} = 0 \iff 1 - \kappa^{(t)^2}\varrho = 0 \iff \underline{J}^{(t)} = \kappa^{(t)} \sum_{\mu\in\Gamma} \Delta x_\mu \underline{\xi}^\mu . \tag{A6}$$

That means $\kappa^{(t+1)} > \kappa^{(t)}$ as long as RECOMI has not terminated. □

References

[1] Hopfield J J 1982 Neural networks and physical systems with emergent computational abilities *Proc. Natl Acad. Sci. USA* **79** 2554–9

[2] Amit D J, Gutfreund H and Sompolinsky H 1985 Storing infinite numbers of patterns in a spin-glass model of neural networks *Phys. Rev. Lett.* **55** 1530–3

[3] Gardner E 1988 The space of interactions in neural network models *J. Phys. A: Math. Gen.* **21** 257–70

[4] Diederich S and Opper M 1987 Learning of correlated patterns in spin-glass networks by local learning rules *Phys. Rev. Lett.* **58** 949–52

[5] Krauth W and Mézard M 1987 Learning algorithms with optimal stability in neural networks *J. Phys. A: Math. Gen.* **20** L745–52

[6] Anlauf J K and Biehl M 1989 The AdaTron: an adaptive perceptron algorithm *Europhys. Lett.* **10** 687–92

[7] Opper M 1988 Learning times of neural networks: exact solution for a PERCEPTRON algorithm *Phys. Rev.* A **38** 3824–6

[8] Ruján P 1993 A fast method for calculating the perceptron with maximal stability *J. Physique I* **3** 277–90

[9] Personnaz L, Guyon I and Dreyfus G 1985 Information storage and retrieval in spin-glass like neural networks *J. Physique Lett.* **46** L359–65

[10] Kanter I and Sompolinsky H 1987 Associative recall of memory without errors *Phys. Rev.* A **35** 380–92

[11] Fletcher R 1987 *Practical Methods of Optimization* (New York: Wiley)