J. Phys. A: Math. Gen. **20** (1987) L745-L752. Printed in the UK

**LETTER TO THE EDITOR**

# Learning algorithms with optimal stablilty in neural networks

Werner Krauth†‡ and Marc Mézard†

† Laboratoire de Physique Théorique de l'Ecole Normale Supérieure, Université de Paris-Sud, 24 rue Lhomond, 75231 Paris Cédex 05, France

‡ Department de Physique de l'Ecole Normale Supérieure, Université de Paris-Sud, 24 rue Lhomond, 75231 Paris Cédex 05, France

Received 19 May 1987

**Abstract.** To ensure large basins of attraction in spin-glass-like neural networks of two-state elements $\xi_i^\mu = \pm 1$, we propose to study learning rules with optimal stability $\Delta$, where $\Delta$ is the largest number satisfying $\Delta \leqslant (\Sigma_j J_{ij}\xi_j^\mu)\xi_i^\mu$; $\mu = 1, \ldots, p$; $i = 1, \ldots, N$ (where $N$ is the number of neurons and $p$ is the number of patterns). We motivate this proposal and provide optimal stability learning rules for two different choices of normalisation for the synaptic matrix $(J_{ij})$. In addition, numerical work is presented which gives the value of the optimal stability for random uncorrelated patterns.

In the last few years, spin-glass models of neural networks have evolved into an active field of research. Much effort has been invested towards the understanding of the Hopfield model (Hopfield 1982) and its generalisations (see recent reviews such as those by Amit and Sompolinsky cited by van Hemmen and Morgenstern (1987)).

These models consist of a network of $N$ neurons (taken to be two-state elements $S_i = \pm 1$) connected to each other through a synaptic matrix $(J_{ij})$. The network evolves in time according to a given dynamical rule, often taken to be a zero-temperature Monte Carlo process:

$$S_i(t+1) = \operatorname{sgn}\left(\sum_j J_{ij} S_j(t)\right). \tag{1}$$

This is the rule we will adopt in the following.

So far the interest in neural networks has been mainly focused on their properties of associative memories. This works as follows: in a so-called 'learning phase', the network is taught a number $p$ of 'patterns' $\boldsymbol{\xi}^\mu$, $\mu = 1, \ldots, p$ (each pattern being a configuration of the network $\boldsymbol{\xi}^\mu = \xi_1^\mu, \xi_2^\mu, \ldots, \xi_N^\mu$; $\xi_i^\mu = \pm 1$), i.e. the corresponding information is encoded into the matrix $(J_{ij})$ by means of a given learning algorithm.

In the retrieval phase, the network is started in a certain initial configuration $\boldsymbol{S}$ $(t = 0)$. If this configuration is not too different from one of the patterns, say $\boldsymbol{\xi}^\mu$, it should evolve under the dynamic rule (1) towards a fixed point, which is the pattern itself $\boldsymbol{S}(t = \infty) = \boldsymbol{\xi}^\mu$. We will say then that $\boldsymbol{S}(t = 0)$ lies in the basin of attraction of $\boldsymbol{\xi}^\mu$. A necessary condition for associative memory in this (rather strict) sense is that the patterns be fixed points of (1) (which implies that the system is at least able to recognise the learned patterns). This can be written as

$$\xi_i^\mu = \operatorname{sgn}\left(\sum_j J_{ij}\xi_j^\mu\right) \qquad \mu = 1, \ldots, p;\ i = 1, \ldots, N \tag{2a}$$

0305-4470/87/110745+08$02.50  

or, equivalently, as

$$0 < \Delta \leqslant \left( \sum_j J_{ij} \xi_j^\mu \right) \xi_i^\mu \qquad \mu - 1, \ldots, p; \; i - 1, \ldots, N. \tag{2b}$$

An important problem in the context of associative memory is to devise learning rules which lead to large memory capacities, i.e. models whose basins of attraction are as large as possible. This is a difficult problem of 'phase-space gardening' which is the inverse problem of the spin-glass one. So far the only proposed rules (Gardner *et al* 1987, Poeppel and Krey 1987) are iterative improvement methods on the matrix $(J_{ij})$: if a given configuration does not converge towards the pattern, one tries to modify $(J_{ij})$ in order to ensure this convergence. Obviously, in order to dig a basin, one must scan a large part of the configurations of the basin and this is very time consuming (the number of configurations which differ in $k$ bits from a pattern grows like $N^k/k!$).

In view of this difficulty, we propose in the present letter to study instead a 'poor man's version' of this problem: the network should have optimal stability $\Delta$. As we shall see, this enables one to guarantee at least a certain minimal size of the basins of attraction; in addition, we will be able to solve this simplified problem, i.e. to provide learning algorithms which compute synaptic couplings resulting in optimal stability of the network.

A network with the dynamical rule (1) is invariant under a rescaling of the $J_{ij}$, and our criterion makes sense only if one has chosen a certain scale for these quantities. Let us assume, therefore, that the synaptic connections satisfy

$$|J_{ij}| \leqslant 1/\sqrt{N} \qquad i, j = 1, \ldots, N \tag{3}$$

and, further, that one starts from an initial configuration $\boldsymbol{S}(t=0)$ which coincides in all but a number $\delta$ of bits (components) with a pattern $\boldsymbol{\xi}^\alpha$. Conditions (1)–(3) then ensure that

$$S_i(t=1) = \operatorname{sgn}\left( \sum_j J_{ij} S_j(t=0) \right) = \operatorname{sgn}\left( \sum_j J_{ij} \xi_j^\alpha \right) = \xi_i^\alpha \tag{4}$$

provided

$$\delta \leqslant \Delta \sqrt{N}/2. \tag{5}$$

The inequality (5) motivates our strategy: the better the stability $\Delta$ of the network, the larger is the size of the region which can be recognised by the network *in one time step*. We will proceed on the assumption that the size of the *whole* basins of attraction of the network will then also be larger. In the absence of analytical methods to calculate basins of attraction, a detailed study of this assumption will require extensive numerical simulations, which we leave for future work. It is to be noted that our criterion is too crude to distinguish the details of the dynamical rules (parallel and sequential updating processes lead to the same result (5)) while it is sensitive to different choices of the normalisation on the synaptic matrix, which will be discussed later.

In the following we will not assume that the matrix $(J_{ij})$ is symmetric. The inequalities (2) then decouple into $N$ systems, each of which states the constraints on one row vector of $(J_{ij})$. On row $i$, the stability condition can therefore be written as

$$0 < \Delta_i \leqslant \boldsymbol{J}_i \cdot \boldsymbol{\eta}_i^\mu \qquad \mu = 1, \ldots, p \tag{6}$$

where $\boldsymbol{J}_i$ is the $i$th row vector of $(J_{ij})$, and where the $\boldsymbol{\eta}_i^\mu$ are defined by $\boldsymbol{\eta}_i^\mu = \xi_i^\mu \boldsymbol{\xi}^\mu$ if self-interactions ($J_{ij} \neq 0$) are allowed and $\boldsymbol{\eta}_i^\mu = \xi_i^\mu(\xi_1^\mu, \ldots, \xi_{i-1}^\mu, \xi_{i+1}^\mu, \ldots, \xi_N^\mu)$ otherwise.

We will not distinguish the two possibilities in the following and will treat $\eta_i^\mu$ as a vector with $N$ components which will also be called a 'pattern' and whose row index $i$ will generally be dropped.

We now treat the problem of computing the synaptic strengths of a network with optimal stability $\Delta$, given the normalisation (3). This can easily be formulated as a linear program in the sense of optimisation theory (cf Papadimitriou and Steiglitz 1982). There are $N+1$ variables $(J_1, J_2, \ldots, J_N, \Delta)$ which must satisfy the set of linear inequalities

$$\begin{aligned} &\sum_i J_i \eta_i^\mu - \Delta \geqslant 0 \qquad \mu = 1, \ldots, p \\ &-1/\sqrt{N} \leqslant J_i \leqslant 1/\sqrt{N} \qquad i = 1, \ldots, N \\ &\Delta \geqslant 0 \end{aligned} \tag{7}$$

and the objective function one wants to maximise is just $\Delta$. A feasible solution of (7) is $\boldsymbol{J} = \boldsymbol{0}$, $\Delta = 0$. Therefore, an optimal solution exists; it can be computed using, e.g., the simplex algorithm (cf Papadimitriou and Steiglitz 1982). If the optimal solution is stable ($\Delta > 0$), it will satisfy $\max_j |J_j| = 1/\sqrt{N}$.

For an actual computation, it is advantageous to start from a dual formulation of (7) (cf Papadimitriou and Steiglitz 1982), in which the special form of the inequalities (7) can be used to obtain an initial basic feasible solution. It seems possible, in addition, that more sophisticated methods of combinatorial optimisation can be brought to bear on this problem to increase the speed of the learning procedure and to make efficient use of the correlations between the $\eta_i$ in different rows of the matrix $(J_{ij})$.

Normalisations different from (3) may be of importance, in particular those which allow $J_{ij}$ to take on discrete values only such as $J_{ij} = \pm 1, 0$. Finding optimal stability networks with these normalisations seems, however, to be a more complicated problem. We have rather, in addition to (3), treated the case where the Euclidean norm is fixed: $|\boldsymbol{J}| = 1$. This problem has an interesting geometrical interpretation, in the light of which other, widely used, learning rules can be understood. The problem:

$$\begin{aligned} &\text{maximise } \Delta > 0, \text{ such that } \sum_i J_i \eta_i^\mu - \Delta \geqslant 0 \qquad \mu = 1, \ldots, p \\ &|\boldsymbol{J}| = 1 \end{aligned} \tag{8}$$

corresponds, in a geometrical picture, to finding the symmetry axis $\boldsymbol{J}$ of the most pointed cone enclosing all the vectors $\boldsymbol{\eta}^\mu$ (note that $|\boldsymbol{\eta}^\mu| = \sqrt{N}$, $\mu = 1, \ldots, p$). The patterns for which the inequalities (8) are tight come to lie on the border of the cone. This is a simple geometrical problem but it transpires that finding an algorithm which solves it in a space of large dimension is not completely trivial. As a first algorithm one might choose for $\boldsymbol{J}$ the unit vector in the direction of the weighted centre of the $\boldsymbol{\eta}^\mu$. This, precisely, is Hebb's learning rule which is used in the Hopfield model. Clearly it has no reason to be optimal and should perform badly when some of the patterns $\boldsymbol{\xi}^\mu$ are correlated, explaining a well known phenomenon. A different algorithm, the pseudoinverse method, has been proposed by Personnaz *et al* (1985) (cf also Kanter and Sompolinsky (1987)). In this case a vector $\boldsymbol{J}$ is sought, such that $\boldsymbol{J} \cdot \boldsymbol{\eta}^\mu = 1$, $\mu = 1, \ldots, p$. $\boldsymbol{J}$ is thus the symmetry axis of the cone, on whose border all the patterns are situated. Such a cone exists if the patterns are linearly independent (so that $p \leqslant N$ is a necessary condition). The pseudoinverse method does not result in an optimal stability $\Delta$ although it gives good results for a small number of uncorrelated patterns.

To determine a synaptic matrix with optimal stability, we present now an iterative method which is based on the perceptron-type algorithm proposed recently by Diederich and Opper (1987). Consider the following minimum-overlap learning rule, which proceeds in a finite number of time steps $t = 0, \ldots, M$, provided a solution of (8) (with $\Delta > 0$) exists.

At time $t = 0$, set $\boldsymbol{J}^{(0)} = \boldsymbol{0}$ (*tabula rasa*).

At $t = 0, 1, \ldots$, determine a pattern $\boldsymbol{\eta}^{\mu(t)}$ that has minimum overlap with $\boldsymbol{J}^{(t)}$:

$$\boldsymbol{J}^{(t)} \cdot \boldsymbol{\eta}^{\mu(t)} = \min_{\nu=1,\ldots,p} \{\boldsymbol{J}^{(t)} \cdot \boldsymbol{\eta}^{\nu}\} \tag{9}$$

and if

$$\boldsymbol{J}^{(t)} \cdot \boldsymbol{\eta}^{\mu(t)} \leqslant c \qquad (c \text{ is a fixed positive number}) \tag{10}$$

use it to update $\boldsymbol{J}^{(t)}$ by

$$\boldsymbol{J}^{(t+1)} = \boldsymbol{J}^{(t)} + (1/N)\boldsymbol{\eta}^{\mu(t)} \tag{11}$$

or if

$$\boldsymbol{J}^{(t)} \cdot \boldsymbol{\eta}^{\mu(t)} > c \qquad (t = M)$$

renormalise $\boldsymbol{J}^{(m)}$ to unity
then stop.

The stability $\Delta_c$ determined by this algorithm is

$$\Delta_c = \min_{\nu=1,\ldots,p} \{\boldsymbol{J}^{(M)} \cdot \boldsymbol{\eta}^{\nu}\}/|\boldsymbol{J}^{(M)}| \geqslant c/|\boldsymbol{J}^{(M)}|. \tag{12}$$

This algorithm differs from that of Diederich and Opper in two points. We allow $c$ to vary instead of taking $c = 1$ (in fact we shall see that the optimal solution is obtained for $c \gg 1$) and among the patterns which satisfy (10) we choose the one which has the minimal overlap (9) instead of updating sequentially.

We now present three results, as follows.

(i) The minimal-overlap algorithm stops after a finite number $M$ of time steps provided a stable optimal solution of (8) exists.

(ii) If $\Delta_{\text{opt}}$ is the stability of the optimal solution of (8) then $\Delta_c$ satisfies

$$\Delta_c \leqslant \Delta_{\text{opt}} \leqslant A\Delta_c \tag{13}$$

where $A$ is a performance guarantee factor which can be measured:

$$A = |\boldsymbol{J}^{(M)}|^2 N/cM \tag{14}$$

and which satisfies

$$1 \leqslant A \leqslant 2 + 1/c. \tag{15}$$

These first two results are simple consequences of a perceptron-type convergence theorem which we shall sketch in appendix 1. They also apply to the algorithm of Diederich and Opper (DO) for which we have thus obtained the performance guarantee $\Delta_{\text{DO}} \geqslant \frac{1}{3}\Delta_{\text{opt}}$.

(iii) For the minimum-overlap algorithm we have the much stronger result:

$$\text{for } c \to \infty \qquad A \to 1 \qquad \text{so that} \qquad \Delta_c \to \Delta_{\text{opt}} (c \to \infty). \tag{16}$$

The proof of (16) is somewhat more complicated; it may be found in appendix 2.

The two algorithms we have presented in this letter clearly work whatever the correlations between patterns. In order to test them and to provide a convenient

reference to other learning rules, we have performed simulations on random patterns for which each of the $\eta_i^\mu$ is $\pm 1$ with equal probability. (With respect to the original network this means that we have taken the diagonal of the synaptic matrix $(J_{ij})$ equal to 0. Nevertheless we keep on denoting by $N$ the total number of components of each $\boldsymbol{\eta}$.) In our simulations we recorded the obtained stabilities $\Delta$ for both algorithms according to each normalisation: $\Delta_1$ rescaled with $\max_j|J_j|\sqrt{N}$ and $\Delta_2$ rescaled with $|\boldsymbol{J}|$. The results are presented in figure 1 for a storage ratio $\alpha = p/N = 0.5$. The results for $N = 80$, $p = 40$, show, e.g., that after termination there can typically be at least 3.3 wrong bits in an initial state $\boldsymbol{S}(t=0)$ to guarantee convergence to a pattern in one time step using the linear program algorithm, while the corresponding number for the minimum-overlap algorithm with $c = 10$ is 1.7 wrong bits.

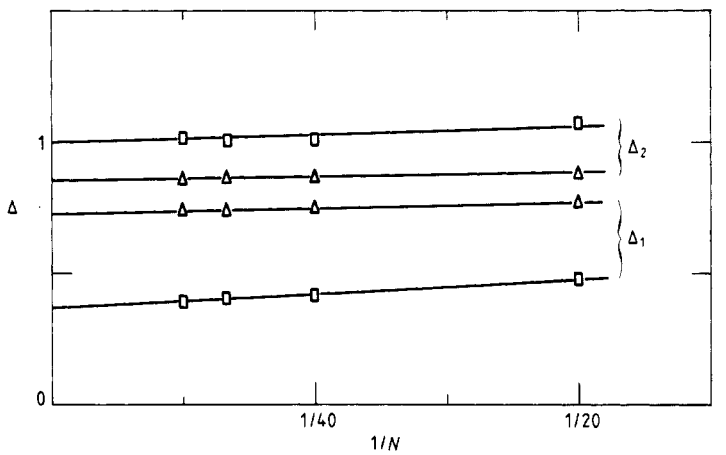


**Figure 1.** Stabilities $\Delta_1$ and $\Delta_2$ found by the two algorithms in the storage of $p = N/2$ uncorrelated random patterns, with $N$ between 20 and 80. The triangles are the results of the simplex and the squares are the results of the minimal-overlap algorithm with $c = 1$. The upper points give $\Delta_2$. Typical averages over 100 samples have been taken for each value of $N$. Lines are guides for the eye; error bars are of the size of the symbols.

For random patterns, the optimal value $\Delta_{\text{opt}}$ as a function of $\alpha$ for $N \to \infty$ has recently been calculated (Gardner 1987). It is the solution of

$$1/\alpha = \int_{-\Delta_{\text{opt}}}^{\infty} \frac{\mathrm{d}t}{\sqrt{2\pi}} \exp(-\tfrac{1}{2}t^2)(t+\Delta_{\text{opt}})^2. \tag{17}$$

Using the formulae (13) and (14), we calculated (with $c = 10$) upper and lower bounds on the value of $\Delta_{\text{opt}}$ which, after statistical averaging, could be extrapolated to $N \to \infty$. The results confirm Gardner's replica calculations, as shown in figure 2. In the large $N$ limit one can store up to $2N$ random uncorrelated patterns (Venkatesh 1986, Gardner 1987).

Finally, we want to mention a possible extension of our second algorithm and we explain it in analogy to the Hopfield model for which it has been shown that only a number of $N/2 \log N$ random patterns can be stored if one requires stability $\Delta > 0$ (Weisbuch and Fogelman-Soulie 1985), while if one allows a small fraction of wrong bits in the retrieved state then the capacity is $p = 0.14\,N$ (Amit *et al* 1985). A similar situation could occur here: it might be sensible to allow a small number of wrong bits

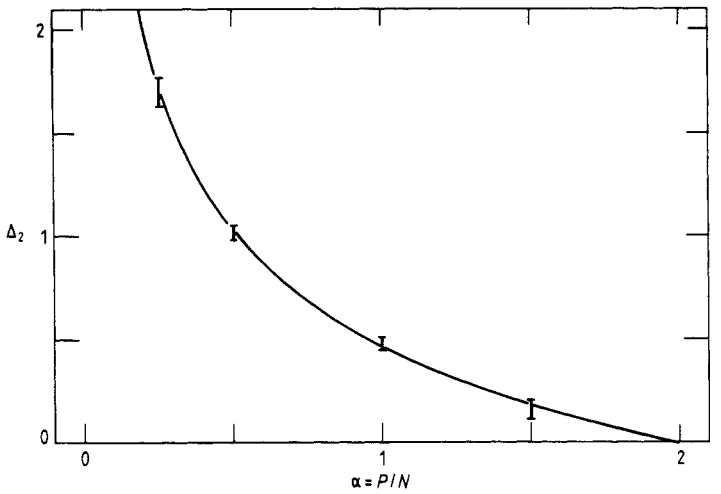


**Figure 2.** Asymptotic value of the optimal stability $\Delta_2$ for the storage of uncorrelated random patterns in the large-$N$ limit as a function of $\alpha = p/N$. The numerical results have been obtained with the minimal-overlap algorithm with $c = 10$. The error bars take into account the uncertainty on the value of $\Delta_{opt}$ due to the fact that $c$ is not infinite (using the bounds (13)), the statistical errors found in averaging over about 100 samples for each size $N$, and a subjective estimate of the uncertainty of the extrapolation to $N \to \infty$. The curve is the prediction (17).

in order to enlarge the size of the basins of attraction (cf Gardner and Derrida (1987) for an analytical approach to this problem). Preliminary work indicates that quite successful methods might be conceived, using a combination of the minimal-overlap method and a simulated annealing method (Kirkpatrick *et al* 1983) with an energy function of the type $E = -\Sigma_\mu \theta(\boldsymbol{J} \cdot \boldsymbol{\eta}^\mu - \Delta)$. In this case the elementary moves can be those of (9)-(11) but a move is accepted only with a certain probability which depends on the change in this energy for the proposed move. It will certainly be interesting to understand how the storage capacities of uncorrelated patterns can be improved with such a method allowing a small number of errors.

It is a pleasure to thank B Derrida, E Gardner, N Sourlas and G Toulouse for stimulating discussions.

## Appendix 1

We prove the convergence of the perceptron-type algorithms and provide bounds on their performance, provided there exists one stable solution. The idea of the proof follows Diederich and Opper (1987).

We assume that there exists an optimal vector $\boldsymbol{J}^*$ such that

$$\begin{aligned} &\boldsymbol{J}^* \cdot \boldsymbol{\eta}^\mu \geqslant c \qquad \mu = 1, \ldots, p \\ &|\boldsymbol{J}^*| = c/\Delta_{opt}. \end{aligned} \tag{A1.1}$$

After $M$ updates with the algorithm (9)-(11), assuming that the pattern $\boldsymbol{\eta}^\mu$ has been used $m^\mu$ times for updating ($\Sigma_\mu m^\mu = M$), one has

$$(M/N)c \leqslant (1/N) \sum_\mu m_\mu \boldsymbol{J}^* \cdot \boldsymbol{\eta}^\mu = \boldsymbol{J}^* \cdot \boldsymbol{J}^{(M)} \leqslant (c/\Delta_{opt})|\boldsymbol{J}^{(M)}|. \tag{A1.2}$$

On the other hand an upper bound on $|J^{(M)}|$ is easily provided by

$$|J^{(t+1)}|^2 - |J^{(t)}|^2 = (2/N)J^{(t)} \cdot \eta^{\mu(t)} + 1/N \leqslant (1/N)(2c+1) \tag{A1.3}$$

which gives

$$|J^{(M)}| \leqslant [M/N(2c+1)]^{1/2}. \tag{A1.4}$$

Therefore the algorithm converges after a bounded number of steps $M$

$$M \leqslant (2c+1)N/\Delta_{\text{opt}}^2 \tag{A1.5}$$

and gives a stability

$$\Delta \geqslant c/|J^{(M)}| \geqslant \Delta_{\text{opt}}(M/N)c/|J^{(M)}|^2 = \Delta_{\text{opt}}/A \tag{A1.6}$$

where $A$ is defined in (14). Furthermore $A$ can be bounded; from (A1.4) and (A1.5) we obtain

$$A = |J^{(M)}|^2 N/cM \leqslant (2c+1)/c = 2 + 1/c. \tag{A1.7}$$

## Appendix 2

To prove (16) we assume again that there exists an optimal solution $J^*$ which satisfies (A1.1). We decompose $J^{(t)}$:

$$\begin{aligned} &J^{(t)} = a(t)J^* + K^{(t)} \\ &K^{(t)} \cdot J^* = 0 \end{aligned} \tag{A2.1}$$

and reason as in appendix 1, but separately on $K^{(t)}$ and $a(t)$.

In the minimal-overlap algorithm, $\eta^{\mu(t)}$ always has a negative projection on $K^{(t)}$:

$$K^{(t)} \cdot \eta^{\mu(t)} \leqslant 0 \tag{A2.2}$$

since otherwise the condition

$$\min_{\mu}\{(J^* + uK(t)) \cdot \eta^{\mu(t)}/|J^* + uK(t)|\} \leqslant \Delta_{\text{opt}} \tag{A2.3}$$

for all $u$ would be violated. As in (A1.3), we can use (A2.2) to show

$$|K^{(t)}| \leqslant \sqrt{t/N}. \tag{A2.4}$$

If learning stops after $M$ time steps, $a(M-1)$ can be bounded as follows:

$$J^{(M-1)} \cdot \eta^{\mu(M-1)} = a(M-1)J^* \cdot \eta^{\mu(M-1)} + K^{(M-1)} \cdot \eta^{\mu(M-1)} < c \tag{A2.5}$$

which yields

$$a(M-1) < 1 + \sqrt{M}/c. \tag{A2.6}$$

The learning rule (11) ensures that $a(M)$ differs little from $a(M-1)$. In fact

$$a(M) \leqslant 1 + \sqrt{M}/c + \Delta_{\text{opt}}/\sqrt{N}c. \tag{A2.7}$$

Equations (A2.4) and (A2.7) can now be combined to bound $|J^{(M)}|$ and using (A1.5) ($M$ grows at most linearly with $c$) we obtain

$$c/\Delta = |J^{(M)}| \to c/\Delta_{\text{opt}} (c \to \infty) \tag{A2.8}$$

which implies the result (16).

Precise bounds and finite $c$ corrections can be obtained using the strategy of appendix 1. They show that the relative precision on $\Delta$ is at least of the order of $1/\sqrt{c}$ for $c$ large. In our numerical simulations we have found a precision which improved rather like $1/c$.

## References

Amit D, Gutfreund H and Sompolinsky H 1985 *Phys. Rev. Lett.* **55** 1530
Diederich S and Opper M 1987 *Phys. Rev. Lett.* **58** 949
Gardner E 1987 *Preprint* Edinburgh 87/395
Gardner E and Derrida B 1987 to be published
Gardner E, Stroud N and Wallace D J 1987 *Preprint* Edinburgh 87/394
Hopfield J 1982 *Proc. Natl Acad. Sci. USA* **79** 2554
Kanter I and Sompolinsky H 1987 *Phys. Rev.* A **35** 380
Kirkpatrick S, Gelatt C D Jr and Vecchi M P 1983 *Science* **220** 671
Papadimitriou D and Steiglitz K 1982 *Combinatorial Optimization: Algorithms and Complexity* (Englewood Cliffs, NJ: Prentice Hall)
Personnaz L, Guyon I and Dreyfus J 1985 *J. Physique Lett.* **16** L359
Poeppel G and Krey U 1987 *Preprint*
van Hemmen L and Morgenstern I 1987 *Lecture Notes in Physics* vol 275 (Berlin: Springer)
Venkatesh S 1986 *Proc. Conf. on Neural Networks for Computing, Snowbird, Utah*
Weisbuch G and Fogelman-Soulie F 1985 *J. Physique Lett.* **46** L263