Phase transitions in the generalization behaviour of multilayer neural networks

1995 J. Phys. A: Math. Gen. 28 4515

(http://iopscience.iop.org/0305-4470/28/16/010)

J. Phys. A: Math. Gen. 28 (1995) 4515–4531. Printed in the UK

Phase transitions in the generalization behaviour of multilayer neural networks

B Schottky

Institut für Physik II, Universität Regensburg, Universitätsstrasse 31, D-93040 Regensburg, Germany

Received 27 March 1995

Abstract. We study the generalization ability of multilayer neural networks with tree architecture for the case of random training sets. The first layer consists of K spherical perceptrons with binary output. A boolean function B computes the final output from the K values produced by the first layer. We first calculate the learning behaviour of Gibbs learning in the case of learnable rules, where teacher and student have the same architecture and the boolean function B is permutation symmetric with respect to the hidden units. In the asymptotic case of high loading $\alpha \to \infty$ (as usual α is the loading parameter) we find that the generalization error vanishes in the same way for all B; the reason being is that there are two effects cancelling each other. In the opposite limit for small α we find qualitatively different behaviour, i.e. some networks undergo a phase transition. We show how these differences in the behaviour depend on certain characteristics of the boolean function B.

We then study the Bayes algorithm, which generalizes according to the majority decision of a certain ensemble of machines, and find that the parity function plays a special role: if the teacher is a parity-machine there always exists a single student parity-machine that generalizes as well as the Bayes algorithm.

1. Introduction

The learning behaviour of a single-layer perceptron has been investigated in detail using the method introduced by Gardner [1, 2]. One can also apply this method to multilayer networks. Different architectures of multilayer neural networks have recently been investigated using different methods where the second layer was fixed to be a committee- or parity-machine (i.e. the final output is given by the majority vote of the hidden nodes or by the product of the hidden node values, respectively), while the first layer consists of binary or spherical perceptrons with overlapping or non-overlapping receptive fields [3–8].

In this paper we concentrate on the case of tree architecture (i.e. non-overlapping receptive fields) with spherical perceptrons in the first layer and an arbitrary boolean function B which computes the final output from the values the first layer produced. We call the boolean function B the *decision function*. The behaviour of tree architectures is simpler than that of architectures with overlapping receptive fields, where additional transitions occur since the subperceptrons of student and teacher can be assigned to one another in different ways [3].

The storage-capacity problem for two-layer architectures with an arbitrary boolean function in the second layer has been investigated in [9] with a replica symmetric ansatz, looking in detail only on the AND-machine. In a recent paper [10] the storage capacity of two-layer networks with $K = 3$ and $K = 5$ hidden nodes and arbitrary but fixed weights in

0305-4470/95/164515+17$19.50

the second layer (which is of course different from the case of an arbitrary boolean function in the second layer) has been investigated by computer simulations. The difficulty is the necessity of replica symmetry breaking (RSB) for the storage-capacity problem [11–13]. The validity of the replica symmetric calculation for the AND-machine as in [9] is discussed in [14].

For the generalization problem and learnable rules a replica symmetric ansatz was proven to lead to correct results concerning the *thermodynamically stable* phase in examples where the phase space was disconnected as in the cases considered here; see [15] for the $K = 2$ parity-machine and for more detail see [16] where the reversed wedge perceptron has recently been investigated as a toy multilayer neural network. Replica symmetry breaking also occurs in other regimes of the phase space which are not thermodynamically stable [15, 16]. In our calculations for the behaviour of Gibbs and Bayes learning, where we are only interested in the thermodynamic stable phase, we use a replica symmetric ansatz. An investigation of RSB is in progress.

As already known, the committee- and the parity-machines show qualitatively different behaviour [4, 7]: the committee-machine starts at finite α with non-trivial generalization, whereas the parity-machine starts with non-trivial generalization only from a critical value α_c and undergoes a phase transition of first or second order, depending on the number K of hidden nodes.

A general calculation for networks with tree architecture allows us to determine the characteristics of the boolean decision function B which are responsible for this different behaviour, i.e. the presence or absence of a phase transition.

This paper is organized as follows. In section 2 we introduce the architectures considered. In section 3 we introduce the corresponding entropy for the generalization problem for Gibbs learning using the replica-symmetric ansatz. Our result also includes the case of mismatched architectures with any decision functions for the student and teacher, respectively. As order parameters we introduce the overlaps between the subperceptrons. In section 4 we calculate the generalization error as a function of the overlaps. From section 5 onwards we restrict ourselves to learnable rules, where student and teacher have the same boolean function B. As already mentioned, B is assumed to be permutation symmetric and the behaviour can then be described by one order parameter q, fixing the overlap of two random representants of the phase space. For this reduced class of networks we calculate the learning behaviour in the limits $q \to 1$ and $q \to 0$ in section 6 and 7, respectively. In section 8 we present results for $K = 4$ showing the variety of behaviour occurring. The Bayes algorithm is investigated in section 9 and in section 10 we give a conclusion.

2. The architecture

We consider two-layer neural networks with non-overlapping receptive fields. The first layer consists of K spherical perceptrons, each with M binary input units and one binary 'hidden output' neuron. In the second layer the final output is computed from these K values by a boolean function B.

The K subperceptrons of the first layer are

$$w_1, w_2, \ldots, w_K \qquad |w_l|^2 = M. \tag{1}$$

The input vector ξ is given by

$$\xi = (\xi_1, \xi_2, \ldots, \xi_K) \tag{2}$$

where every vector ξ_l consists of M numbers $\in \{\pm 1\}$.

The binary output values of the hidden neurons are

$$\sigma_l = \mathrm{sgn}(h_l) \qquad l = 1, \ldots, K \tag{3}$$

where $h = (h_1, \ldots, h_K)$ is the local field with

$$h_l = \frac{1}{\sqrt{M}} w_l \cdot \xi_l \qquad l = 1, \ldots, K. \tag{4}$$

The vector $\{\sigma_l\}_{l=1,\ldots,K} = (\sigma_1, \ldots, \sigma_K)$ is also called the 'internal representation' (IR). The final output σ is given by

$$\sigma = B(\{\sigma_l\}) \tag{5}$$

where B is an arbitrary K-digit boolean function.

3. The entropy for the generalization problem

In a general ansatz the student and the teacher may have different boolean decision functions, B_s for the student and B_t for the teacher. The number K of hidden nodes are assumed to be the same. The perceptrons of the first layer are w_l^t for the teacher and w_l^s for the student. We are interested in Gibbs learning with zero temperature. The training set consists of p randomly chosen patterns and the corresponding answers of the teacher. The class of all students classifying this training set without error is called the version space $\mathcal{V}$. For Gibbs learning, the student is one randomly chosen member of $\mathcal{V}$. To obtain the typical behaviour one calculates, as usual, the averaged logarithm of the relative phase-space volume:

$$S = \langle \langle \ln V \rangle_{\{\xi\}_p} \rangle_{\mathbf{B}} \tag{6}$$

with V given by

$$V = \frac{1}{L} \int \prod_{l=1}^{K} \mathrm{d}w_l^s \prod_l \delta(M - |w_l^s|^2) \prod_{\mu=1}^{p} \delta\left[B_t(\{\mathrm{sgn}(w_l^t \cdot \xi_l^\mu)\}), B_s(\{\mathrm{sgn}(w_l^s \cdot \xi_l^\mu)\})\right] \tag{7}$$

where the normalization constant L is

$$L = \int \prod_{l=1}^{K} \mathrm{d}w_l^s \prod_l \delta(M - |w_l^s|^2). \tag{8}$$

So we have $0 \leqslant V \leqslant 1$ which implies that S is negative or zero. The second δ in (7) is a Kronecker delta. The average in (6) is taken over all possible teacher networks **B** (with fixed teacher function B_t) and all (uniformly distributed) binary training sets. As usual the loading parameter α is defined as

$$\alpha = \frac{p}{N} \tag{9}$$

where p is the size of the training set and $N = KM$ the size of a total input vector. In the following we use the abbreviations

$$\Delta_t^\sigma(\{\sigma_l\}) = \begin{cases} 1 & \text{if } B_t(\{\sigma_l\}) = \sigma \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

$$\Delta_s^\sigma(\{\sigma_l\}) = \begin{cases} 1 & \text{if } B_s(\{\sigma_l\}) = \sigma \\ 0 & \text{otherwise.} \end{cases} \tag{11}$$

In the thermodynamic limit $N \to \infty$ we obtain the entropy S in the saddle-point approximation using the replica method, where replica symmetry is assumed. As order

parameters we introduce the overlap of corresponding subperceptrons between teacher and student:

$$r_l = \frac{w_l^t \cdot w_l^s}{|w_l^t||w_l^s|} \tag{12}$$

and between two students of the replicated phase space,

$$q_l = \frac{w_l^{s,\beta} \cdot w_l^{s,\gamma}}{|w_l^{s,\beta}||w_l^{s,\gamma}|} \tag{13}$$

where β and γ are two (different) replica indices. With the replica method, see e.g. [17], we obtain

$$S = \operatorname*{extr}_{\{r_l,q_l\}} \left\{ \frac{1}{K} \sum_{l=1}^{K} \left[\frac{1}{2} \ln(1-q_l) + \frac{1}{2} \frac{q_l - r_l^2}{1-q_l} \right] - \alpha W(\{r_l, q_l\}) \right\} \tag{14}$$

with

$$W(\{r_l, q_l\}) = -\int \prod_{l=1}^{K} (\mathrm{D}t_l \, \mathrm{D}s_l) \sum_{\sigma=\pm 1} \left[\operatorname*{Tr}_{\{\sigma_l\}} \Delta_t^{\sigma}(\{\sigma_l\}) \prod_l F_t(r_l, t_l, \sigma_l) \right]$$
$$\times \ln \left[\operatorname*{Tr}_{\{\sigma_l\}} \Delta_s^{\sigma}(\{\sigma_l\}) \prod_l F_s(r_l, q_l, t_l, s_l, \sigma_l) \right] \tag{15}$$

and

$$F_t(r, t, \sigma) = H(\sigma \gamma_r t) \tag{16}$$
$$F_s(r, q, t, s, \sigma) = H[\sigma(\gamma_1 t + \gamma_2 s)] \tag{17}$$
$$\gamma_r = \sqrt{\frac{r}{1-r}} \qquad \gamma_1 = \sqrt{\frac{r}{1-q}} \qquad \gamma_2 = \sqrt{\frac{q-r}{1-q}} \tag{18}$$

where

$$H(x) = \int_x^{\infty} \mathrm{D}t \qquad \mathrm{D}t = \frac{1}{\sqrt{2\pi}} \mathrm{e}^{-t^2/2} \, \mathrm{d}t. \tag{19}$$

In (15) 'Tr' means the sum over all possible IRs. The condition that the entropy is to be maximized, leads to an expression for the order parameters $r_l(\alpha)$ and $q_l(\alpha)$ as a function of α.

4. The generalization ability for a given overlap

The generalization error is the probability that a randomly chosen question is answered differently by the student and the teacher:

$$\epsilon(\{r_l, q_l\}) = \int \prod_{l=1}^{K} (\mathrm{D}t_l \mathrm{D}s_l) \sum_{\sigma=\pm 1} \left[\operatorname*{Tr}_{\{\sigma_l\}} \Delta_t^{\sigma}(\{\sigma_l\}) \prod_l F_t(r_l, t_l, \sigma_l) \right]$$
$$\times \left[\operatorname*{Tr}_{\{\sigma_l\}} \Delta_s^{-\sigma}(\{\sigma_l\}) \prod_l F_s(r_l, q_l, t_l, s_l, \sigma_l) \right]. \tag{20}$$

Evaluating this expression we obtain a result which is independent of the $\{q_l\}$ as expected: since the overlap of a student to the teacher, and not the overlap between two students, is

the relevant value determining the generalization behaviour, we obtain

$$\epsilon = \epsilon(\{r_l\}) = \left(\frac{1}{2}\right)^K \sum_{\sigma=\pm 1} \left[\operatorname*{Tr}_{\{\sigma_l^t\}} \operatorname*{Tr}_{\{\sigma_l^s\}} \Delta_t^{\sigma}(\{\sigma_l^t\}) \Delta_s^{-\sigma}(\{\sigma_l^s\}) \prod_l \left(1 - \frac{1}{\pi} \arccos(\sigma_l^t \sigma_l^s r_l)\right) \right]. \tag{21}$$

The learning behaviour of the architectures considered is fully determined by (14), (15) and (21). Nevertheless one has to take into account that for $B_t \neq B_s$ replica symmetry breaking is to be expected, so in that case the replica-symmetric ansatz is only an approximation. In the following, however, we consider only learnable problems.

5. Learnable problems with permutation symmetry

We restrict ourselves to the case where the student and teacher networks have the same architecture, which means that $B_t = B_s = B$. Additionally, B is assumed to be permutation symmetric which means that $B(\{\sigma_l\})$ depends only on the number of 1's in the IR $\{\sigma_l\}$, but not on the order. For example, the parity- and the committee-machines are permutation symmetric, whereas the so called ruler-machine (where one node value determines the final output) is not.

For a given K the number N_{ps} of non-equivalent permutation symmetric K-digit boolean functions is

$$N_{ps} = 2^{K-1} + 2^{\lfloor (K-1)/2 \rfloor} - 1 \tag{22}$$

where the trivial case (same output for all internal representations) is not counted. To understand (22) one has to consider that the (permutation symmetric) decision function B is determined by specifying the answers $a_n \in \{\pm 1\}$ on an IR containing n 1's ($n = 0, \ldots, K$). So the bit sequence $a_0 a_1 \cdots a_K$ specifies B. So far we have 2^{K+1} different decision functions, but some of them are isomorphic: considering the sequence $a_0 a_1 \cdots a_K$ one arrives at an isomorphic function by (i) changing every sign of the a_n's, (ii) by inverting this sequence to $a_K \cdots a_1 a_0$ and (iii) by doing (i) and (ii) together. By avoiding these double-counted functions one arrives at (22).

Because all hidden nodes are equal in role, no distinction of the order parameter of the K subperceptrons is necessary. Since student and teacher have the same architecture, the teacher is in the same phase space as all training-error free students. Therefore r and q become identical so that the number of order parameters reduces to one and (14) becomes

$$S = \max_q S(q) = \max_q \left\{ \frac{1}{2} \ln(1-q) + \frac{q}{2} - \alpha W(q) \right\} \tag{23}$$

with

$$W(q) = -\int \prod_l \mathrm{D}t_l \sum_{\sigma=\pm 1} \left[\operatorname*{Tr}_{\{\sigma_l\}} \Delta^{\sigma}(\{\sigma_l\}) \prod_l H(\sigma_l \gamma t_l) \right] \ln \left[\operatorname*{Tr}_{\{\sigma_l\}} \Delta^{\sigma}(\{\sigma_l\}) \prod_l H(\sigma_l \gamma t_l) \right] \tag{24}$$

where

$$\gamma = \sqrt{\frac{q}{1-q}}. \tag{25}$$

$S(q)$ is negative for all q and α, in agreement with definition (6). $W(q)$ can be interpreted in a rather simple way. We consider the version space V with typical overlap q and think

of a single subperceptron with number l. A random pattern has a local field at the lth subperceptron of each member of $\mathcal{V}$ with the distribution

$$h_l = s_l\sqrt{1-q} - t_l\sqrt{q}. \tag{26}$$

Hence, h_l consists of two parts: since all members of $\mathcal{V}$ have typical overlap q among each other, the local field has the *common part* $-t_l\sqrt{q}$ where t_l is a variable due to the random choice of the pattern. This part is the same for all members of $\mathcal{V}$. The part $s_l\sqrt{1-q}$ corresponds to the distribution of the members of $\mathcal{V}$, so it is different for different members of $\mathcal{V}$. Both t_l and s_l are Gaussian random variables with mean zero and variance one.

Now

$$H(\sigma_l\gamma t_l) = \int_{-\infty}^{\infty} \mathrm{D}s_l\Theta\left(\sigma_l\left[s_l\sqrt{1-q} - t_l\sqrt{q}\right]\right) \tag{27}$$

is the relative volume of the version space, where subperceptron l gives answer σ_l to a pattern with common local field $-t_l\sqrt{q}$. So

$$p_\sigma = \operatorname*{Tr}_{\{\sigma_l\}} \Delta^\sigma(\{\sigma_l\}) \prod_l H(\sigma_l\gamma t_l) \tag{28}$$

is the part of $\mathcal{V}$ giving final output σ to a pattern which has common parts of the local field given by the t_l's. So every random pattern divides $\mathcal{V}$ into two parts: one part with relative size p_1 of all networks answering this pattern with $\sigma = 1$, the other part with size $p_{-1} = 1 - p_1$ giving the answer $\sigma = -1$. The entropy w of this distribution is

$$w = -p_1\ln(p_1) - (1-p_1)\ln(1-p_1) \tag{29}$$

with p_σ given by (28) and where the $\mathrm{D}t_l$ integrations in (24) perform the average over the pattern distribution. So $W(q)$ is the average information gain of the system by a pattern answered by the teacher for a given overlap q.

The expression for the generalization error simplifies to

$$\epsilon(q) = \left(\frac{1}{2}\right)^K \sum_{\sigma=\pm 1}\left[\operatorname*{Tr}_{\{\sigma_l^t\}}\operatorname*{Tr}_{\{\sigma_l^s\}} \Delta^\sigma(\{\sigma_l^t\})\Delta^{-\sigma}(\{\sigma_l^s\}) \prod_l \left(1 - \frac{1}{\pi}\arccos(\sigma_l^t\sigma_l^s q)\right)\right]. \tag{30}$$

In the following sections we exclude the uninteresting cases where the boolean function $B(\{\sigma_l\})$ is constant.

6. Behaviour for $q \to 1$

This limit corresponds to $\alpha \to \infty$. To obtain the behaviour of $\epsilon(q)$ one has to consider terms like

$$\left[1 - \frac{1}{\pi}\arccos(q)\right]^{K-m}\left[1 - \frac{1}{\pi}\arccos(-q)\right]^m \to \tag{31}$$

$$\left[1 - (K-m)\frac{1}{\pi}\sqrt{2}\sqrt{1-q}\right]\left[\frac{1}{\pi}\sqrt{2}\sqrt{1-q}\right]^m. \tag{32}$$

Taking the double trace in (30) one has to sum over these expressions, where m is defined as the number of hidden nodes and where $\sigma_l^t \neq \sigma_l^s$ and $\{\sigma_l^t\}$ and $\{\sigma_l^s\}$ are the IRs of the teacher and student, respectively. We are now interested in terms of lowest order in $\sqrt{1-q}$, leading to

$$\epsilon(q) \to n_c\frac{1}{\pi}\sqrt{2}\sqrt{1-q} \tag{33}$$

$$=: c_1\sqrt{1-q} \tag{34}$$

where n_c is defined as follows. Given K hidden nodes there are 2^K possible IRs each of which consists of K bits. Changing one bit in the IR can either lead to a different output or it can leave it unchanged. The number of all of those $K2^K$ possible bitflips which lead to a changing output is called N_c. We then define n_c as

$$n_c = (\tfrac{1}{2})^K N_c. \tag{35}$$

N_c can be called the 'border-regime size', referring to the border between the local field volumes mapped to final output 1 and -1, respectively.

So the only architecture-dependent value determing the asymptotic $\epsilon(q \to 1)$ is n_c. This is obvious, since in the limit of a nearly perfect student the relevant errors (concerning the probability of occurrence) are those where the internal representations of teacher and student differ in only one bit.

We now consider the entropy S in the limit $q \to 1$. First, we expand the function $W(q)$ from (24) for this limit in lowest order of $\sqrt{1-q}$:

$$W(q) \to -n_c \sqrt{\frac{2}{\pi}} \int H(x) \ln H(x)\, dx \sqrt{1-q} \tag{36}$$

$$=: c_2 \sqrt{1-q} \tag{37}$$

where n_c has the meaning introduced above. Hence the same value n_c is relevant for $W(q)$ in the limit $q \to 1$ as for $\epsilon(q)$. The proportionality of $W(q)$ to n_c can be seen as follows. In the limit where student and teacher are nearly the same, almost all networks in the phase space give the same answer σ. The probability p_1 for the answer $-\sigma$ is then proportional to n_c with the same argument as for $\epsilon(q)$: the errors caused by only one differing bit in the hidden nodes are the most likely ones:

$$p_1 = n_c f(q). \tag{38}$$

Thus, p_1 factorizes into the architecture-dependent constant n_c and the architecture-independent function $f(q)$ with $f(q) \to 0$ for $q \to 1$. Inserting this into (29) one obtains

$$w = W(q) = -n_c f(q) \ln(n_c f(q)) - (1 - n_c f(q)) \ln(1 - n_c f(q)) \to -n_c f(q) \ln(f(q)) \tag{39}$$

in lowest order of $f(q) \to 0$. Thus, not only $\epsilon(q)$, see (34), but also $W(q)$ is proportional to n_c.

Now the entropy $S(q)$ has the asymptotic form

$$S(q) \to \frac{1}{2} \ln(1-q) + \frac{q}{2} - \alpha c_2 \sqrt{1-q}. \tag{40}$$

The condition $\partial S / \partial q = 0$ leads to

$$q \to 1 - \frac{1}{(c_2)^2} \frac{1}{\alpha^2}. \tag{41}$$

Substituting this into (34) one has the result

$$\epsilon(\alpha) \to \frac{c_1}{c_2} \frac{1}{\alpha} = -\frac{1}{\sqrt{\pi}} \left[\int H(x) \ln H(x)\, dx \right]^{-1} \frac{1}{\alpha} \approx 0.6246\ldots \left(\frac{1}{\alpha} \right). \tag{42}$$

The asymptotic behaviour of the generalization error in the limit $\alpha \to \infty$ is, therefore, the same for all permutation-symmetric tree architectures. In particular, it is identical to that of the simple perceptron. This is because there are two effects cancelling each other, namely the proportionality of $\epsilon(q)$ and $W(q)$ to n_c for $q \to 1$.

In [18] we found a similar effect for the clock model, where the couplings are complex numbers and the input and output states are given by the Qth roots $e^{2\pi i n/Q}$ of 1 ($n = 0, \ldots, Q-1$). The asymptotic behaviours of $q(\alpha)$, $\epsilon(q)$ and $\epsilon(\alpha)$ for $q \to 1$ ($\alpha \to \infty$) have the same dependence on Q as in the case considered here on n_c. The function $\epsilon(\alpha)$ did not, therefore, depend on Q. The reason being is that Q in [18] plays the role of n_c, since the number of possibilities to move in a wrong classification area by crossing one border is given by Q in the clock model and by n_c for the case considered here. However, the prefactor for $\epsilon(\alpha)$ in the limit $\alpha \to \infty$ is twice as large for the clock model as for the present case: due to the complex couplings, there are twice as many degrees of freedom in [18] which causes the factor two in the asymptotic of $\epsilon(\alpha)$. For the Potts model the degrees of freedom depend on Q and the asymptotic behaviour of $\epsilon(\alpha)$ is not Q-independent [19].

7. Behaviour for $q \to 0$

In the limit $q \to 0$ we also find a number of general results. Expanding the generalization error from (30) for small q we obtain

$$\prod_l \left[1 - \frac{1}{\pi}\arccos(\sigma_l^t \sigma_l^s q)\right] \to \left(\frac{1}{2}\right)^K \left[1 + \frac{2}{\pi} q \sum_l \sigma_l^t \sigma_l^s + \left(\frac{2}{\pi}\right)^2 q^2 \sum_{l_1 > l_2} \sigma_{l_1}^t \sigma_{l_1}^s \sigma_{l_2}^t \sigma_{l_2}^s + \cdots\right] \tag{43}$$

using

$$1 - \frac{1}{\pi}\arccos(x) \overset{x \to 0}{\to} \frac{1}{2} + \frac{1}{\pi}\left(x + \frac{x^3}{6} + \cdots\right). \tag{44}$$

Taking into account that $(\sigma)^m = \sigma$ for odd m, one has to consider terms like

$$\operatorname*{Tr}_{\{\sigma_l^t\}} \operatorname*{Tr}_{\{\sigma_l^s\}} \Delta^\sigma(\{\sigma_l^t\}) \Delta^{-\sigma}(\{\sigma_l^s\}) \sum_{l_1 > \cdots > l_m} \sigma_{l_1}^t \sigma_{l_1}^s \cdots \sigma_{l_m}^t \sigma_{l_m}^s \tag{45}$$

$$= \binom{K}{m} \left(\operatorname*{Tr}_{\{\sigma_l\}} \Delta^\sigma(\{\sigma_l\}) \prod_{i=1}^m \sigma_i\right) \left(\operatorname*{Tr}_{\{\sigma_l\}} \Delta^{-\sigma}(\{\sigma_l\}) \prod_{i=1}^m \sigma_i\right) \tag{46}$$

in performing the double trace from (30). Here we have used the permutation symmetry of B. We now define

$$a_m^\sigma := \left(\frac{1}{2}\right)^K \operatorname*{Tr}_{\{\sigma_l\}} \Delta^\sigma(\{\sigma_l\}) \prod_{i=1}^m \sigma_l \tag{47}$$

where the empty product is defined as one. These a_m^σ are simply the averaged correlations of m node values of an IR. Specifically, a_m^σ is the mth correlation moment of the IRs mapped to σ. Nevertheless, one has to pay attention to the fact that the averaging has to be taken over all IRs, but the correlation values are set to zero for the IR not leading to the output σ:

$$a_m^\sigma = \langle \sigma_1 \cdots \sigma_m \delta[B(\{\sigma_l\}), \sigma] \rangle_{\{\sigma_l\}}. \tag{48}$$

In particular, a_0^σ is the fraction of all IRs leading to output σ. For the coefficients a_m^σ we have the relations

$$a_0^1 + a_0^{-1} = 1 \tag{49}$$

$$a_m^\sigma = -a_m^{-\sigma} \qquad \text{for } m \geqslant 1. \tag{50}$$

The lowest order of q in $\epsilon(q)$ is determined by the first non-vanishing a_m^σ. Defining n as the index corresponding to

$$a_n^\sigma \neq 0 \qquad a_m^\sigma = 0 \qquad \text{for } 1 \leqslant m < n \tag{51}$$

one obtains for $\epsilon(q)$ in lowest order

$$\epsilon(q) \to 2a_0^1 a_0^{-1} - q^n 2\left(\frac{2}{\pi}\right)^n \binom{K}{n} \left(a_n^1\right)^2 \tag{52}$$

$$=: \epsilon_0 - q^n \epsilon_1. \tag{53}$$

To obtain the entropy $S(q)$ in this limit we first consider the behaviour of $W(q)$ from (24). Expanding the integrand we can perform the integration. Taking the trace one obtains in lowest order of q

$$W(q) \to -a_0^1 \ln a_0^1 - a_0^{-1} \ln a_0^{-1} - q^n \frac{1}{2}\left(\frac{2}{\pi}\right)^n \binom{K}{n} \frac{\left(a_n^1\right)^2}{a_0^1 a_0^{-1}} \tag{54}$$

$$=: w_0 - q^n w_1. \tag{55}$$

The a_m^σ are defined as above. For S we have

$$S(q) \to -\frac{q^2}{4} - \alpha(w_0 - q^n w_1). \tag{56}$$

ϵ_0 from (53) is the generalization error when the system simply guesses. w_0 in (55) is the information content which a totally untrained student ($q = 0$) acquires by getting a question answered by the teacher.

To obtain the asymptotic behaviour of $q(\alpha)$ and $\epsilon(\alpha)$ according to (53) and (56) one has to distinguish between three cases dependent on n.

7.1. The case $n = 1$

Looking at (56) for $n = 1$ and $\alpha > 0$, the entropy $S(q)$ is not maximal at $q = 0$ since $\partial S/\partial q|_{q=0} > 0$. So the location of the maximum of $S(q)$ is (at least for small enough q) given by the condition $\partial S/\partial q = 0$, leading to

$$q = 2\alpha w_1. \tag{57}$$

Inserting this into the asymptotic expansion of the generalization error (53), we have

$$\epsilon(\alpha) \to \epsilon_0 - \alpha\epsilon_1 \tag{58}$$

with

$$\epsilon_1 = \frac{8}{\pi^2} K^2 \frac{\left(a_1^1\right)^4}{a_0^1 a_0^{-1}}. \tag{59}$$

Thus, the system has a non-trivial generalization ability for every $\alpha > 0$. We will see that in contrast to this all networks with $n \geqslant 2$ have a non-trivial generalization ability only up from a critical value α_c of the loading parameter. The specific example of $n = 1$ compared to $n \geqslant 2$ is that $a_1^\sigma \neq 0$. This means that in the class of all IRs leading to the final output $\sigma = 1$ (for example) there is a preferred sign of the node values. Either 1 or -1 occurs more often than its opposite value! Hence, one needs a preferred sign in the IRs (for a given σ) to obtain a non-trivial generalization for small α.

7.2. The case $n = 2$

$n \geqslant 2$ means that there is no preferred sign in the IRs which leads to a specific final output σ. We will see that these networks reach no generalization ability up to a certain critical value of the loading parameter α. For small q the entropy $S(q)$ behaves like

$$S(q) \to -\tfrac{1}{4}q^2 - \alpha(w_0 - q^2 w_1) \tag{60}$$

$$= -\alpha w_0 + q^2(\alpha w_1 - \tfrac{1}{4}). \tag{61}$$

This means that up to a critical value of α, namely

$$\bar{\alpha}_c = \frac{1}{4w_1} \tag{62}$$

the entropy $S(q)$ decreases when q increases from zero. Now two cases are possible.

(i) The entropy S has its maximum at $q = 0$ for $\alpha < \bar{\alpha}_c$, so q is zero up to this value of α. At $\alpha = \bar{\alpha}_c$ one obtains a phase transition of second order. With w_1 from (55) it follows that

$$\bar{\alpha}_c = \frac{\pi^2 \left(a_0^1 a_0^{-1}\right)}{4K(K-1)\left(a_2^1\right)^2}. \tag{63}$$

Considering higher orders the result for the generalization error is typically

$$\epsilon(\alpha) - \epsilon_0 \sim (\alpha - \bar{\alpha}_c)^2 \Theta(\alpha - \bar{\alpha}_c). \tag{64}$$

(ii) For $\alpha_c' < \bar{\alpha}_c$, due to the higher-order terms in q, the entropy is for some $q > 0$ higher than for $q = 0$. Then q is zero only up to α_c' undergoing a phase transition of first order to non-trivial generalization behaviour at this value of α.

So for $n = 2$ phase transitions of first and second order are possible, or even both transitions can occur together at different α in the same machine (see the section with the results). The $\bar{\alpha}_c$ from (63) is at least an upper limit for the occurrence of a phase transition.

7.3. The case $n \geqslant 3$

In this case the function $q(\alpha)$ always has a phase transition of first order. This can easily be seen from the expansion of $S(q)$ for $q \to 0$:

$$S(q) \to -\tfrac{1}{4}q^2 - \alpha(w_0 - q^n w_1) \tag{65}$$

$$= q^2 \underbrace{(q^{n-2}\alpha w_1 - \tfrac{1}{4})}_{(*)} - \alpha w_0. \tag{66}$$

The expression $(*)$ is always negative if q is chosen small enough. Therefore, we always obtain an entropy $S(q)$ decreasing as q increases from zero. The maximum of the entropy is, therefore, either at $q = 0$ or at $q \geqslant q_{\min} > 0$. This results in a phase transition of first order at a certain α_c.

7.4. Phase transition for $n = 1$

We have seen that for $n = 1$ the generalization ability increases at once for $\alpha > 0$. This increase can be very small, in fact it is very small in many cases (see examples in the next section). In these cases there is an area of rapidly decreasing generalization error around a specific α. Sometimes there is a jump which means there can also be a phase transition of first order for $n = 1$. In contrast to the cases $n \geqslant 2$, the overlap q jumps at this point from a finite value larger than zero to a higher value.

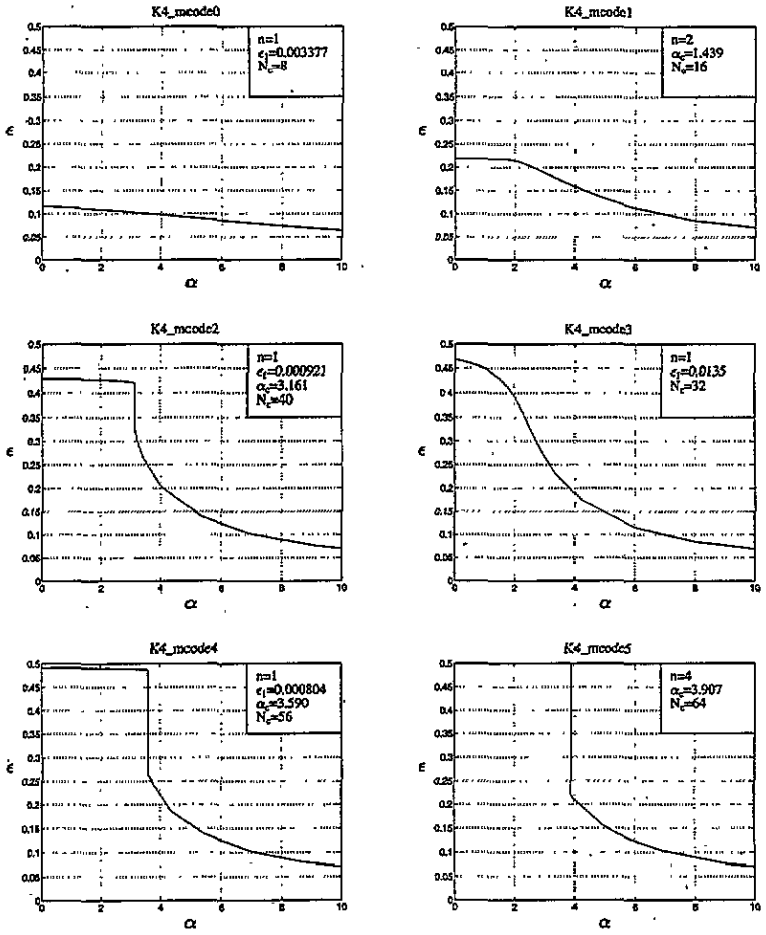

Figure 1. The generalization error ϵ dependent on the loading parameter α for the nine network types with $K = 4$ is shown. The values in the top right-hand corner are the number n of the first non-vanishing correlation moment of the IRs, the slope ϵ_1 of $\epsilon(\alpha)|_{\alpha=0}$ (for $n = 1$), the *phase transition point* α_c *(if existing) and the border-regime size* N_c. *The labelling system of* the network types is explained in the text. For K4_mcode11 there are two α_c's, since there is first a phase transition of second order at $\alpha_c^1 = \bar{\alpha}_c$ from (63) and then a phase transition of first order at α_c^2, with $\alpha_c^1 = 3.084$ and $\alpha_c^2 = 3.269$.

8. Results for $K = 4$

In figure 1 we present the function $\epsilon(\alpha)$ for all networks with $K = 4$. Every network architecture has a number called the 'mcode' which specifies (together with the number K of hidden nodes) the boolean function B in the following way. B is determined by the $K+1$ output values dependent on the number of 1's in an IR. The lth bit (the bit corresponding to the number 2^l, $l = 0, 1, \ldots$) of the mcode is on (=1) if the output for l 1's in an IR is *one, and off* (=0) *otherwise. The output to the* IR *with* K 1's *is fixed to one. This is no* restriction since there is no prior distinction between the outputs 1 and -1. So the range of possible values for the mcode at given K is $0, \ldots, 2^K - 2$, where mcode $= 2^K - 1$ is excluded because it is the trivial function always giving the answer one. Some of these

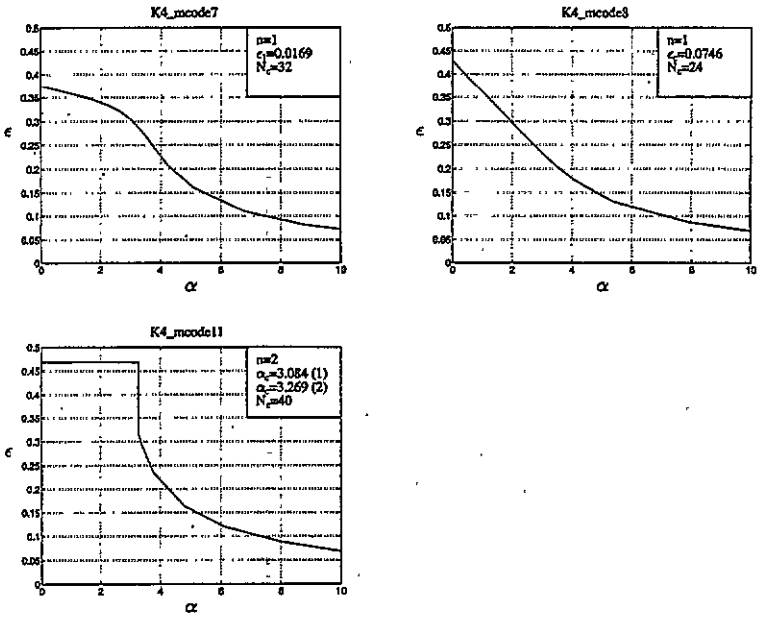

Figure 1. (Continued)

mcodes lead to equivalent machines due to the equivalence of 1 and -1 in an IR, so there are nine different (see (22)) machines for $K = 4$. A machine with i nodes and mcode m is labelled with 'Ki_mcode$m$'. For example the AND-machine (final output is one only if all hidden-node values are one, otherwise the answer is always zero) with $K = 4$ has the label 'K4_mcode0', since $0 = 0 \cdot 2^0 + 0 \cdot 2^1 + 0 \cdot 2^2 + 0 \cdot 2^3$; the parity-machine for $K = 4$ is labelled 'K4_mcode5', since $5 = 1 \cdot 2^0 + 0 \cdot 2^1 + 1 \cdot 2^2 + 0 \cdot 2^3$.

For $K = 4$ there is one machine with $n = 4$ (the parity-machine), two machines with $n = 2$ and six machines with $n = 1$: K4_mcode2 and K4_mcode4 have $n = 1$ but, nevertheless, they undergo a phase transition of first order. This corresponds to the very slow decrease of $\epsilon(\alpha)$ at $\alpha = 0$. K4_mcode1 has $n = 2$ and a phase transition of second order for $\alpha_c = \bar{\alpha}_c$. K4_mcode11 is somehow special: it has $n = 2$ and a phase transition of second order at $\bar{\alpha}_c$ (which is not visible in figure 1, since the decrease of ϵ is very small). However, at a little higher value α_c, it undergoes a phase transition of first order. The calculation accuracy should be good enough so that this is no artificial effect. K4_mcode5 is the parity-machine, all others have $n = 1$ and no phase transition, but K4_mcode3 and K4_mcode7 show an area where the decrease of the generalization error increases rapidly.

For $K > 4$ networks with $n = 2$ often have a high $\bar{\alpha}_c$. They then have a phase transition at a lower α_c of first order.

Heuristically we see that the α_c for the occurrence of a phase transition of first order is roughly determined by N_c, see (35). For small N_c no first-order transition occurs (but eventually there is a second-order transition); the higher the value of N_c, then the higher the value of α_c. For $K = 4$ and $K = 5$ this dependence can be seen in figure 2. Nevertheless this is no 'hard' criterion, only a hint. For example at $K = 6$ there are architectures with lower N_c but higher α_c than others, and there are architectures with $N_c = 192$ undergoing a phase transition of first order but others that do not.

Up to $K = 6$ we found that the parity-machine (which is characterized by $n = K$ and

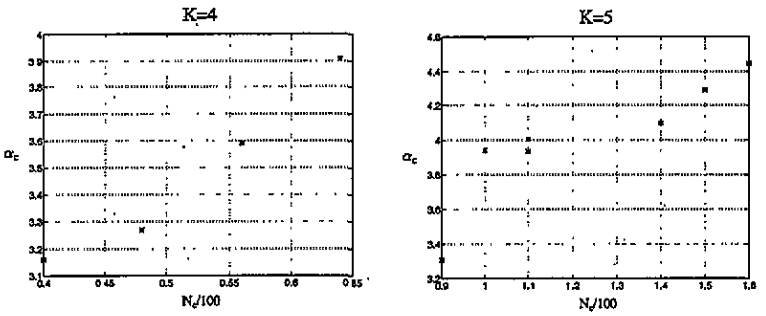

Figure 2. For $K = 4$ and $K = 5$ the dependence of α_c on N_c is shown. All architectures with no phase transition of first order have lower N_c than the beginning of the axis. Nevertheless, there are architectures with a phase transition of second order which have lower N_c.

has the highest N_c for fixed K) has an α_c which is an upper limit of all first order α_c's. We assume that this limit is general for all K, and that for $\alpha > \alpha_c^{\text{parity}}$ all networks are roughly in the asymptotic region ($\alpha \to \infty$). Since $\alpha_c^{\text{parity}} \simeq \ln K / \ln 2$ for large K [7] this is a general statement determing the α from which one good generalization is guaranteed.

It is an interesting fact that for increasing K the case of machines with a phase transition or at least an intermediate regime of rapidly increasing generalization ability becomes the typical case. So the 'aha effect' often seen in real life is also observed in our considerations.

9. The Bayes algorithm

The Bayes algorithm (see e.g. [20]) has the information-theoretic best generalization ability. It responds to a question with the answer which the majority of all perfect students give. If two perfect students, i.e. two members of the version space, have typical overlap q then the Bayes generalization error is given by

$$\epsilon_{\text{Bayes}}(q) = \int \prod_l \mathrm{D}t_l \, \min\left[\operatorname*{Tr}_{\{\sigma_l\}} \Delta^1(\{\sigma_l\}) \prod_l H(\sigma_l \gamma t_l), \operatorname*{Tr}_{\{\sigma_l\}} \Delta^{-1}(\{\sigma_l\}) \prod_l H(\sigma_l \gamma t_l) \right] \tag{67}$$

with γ given by (25). The 'min' in (67) means that the error probability is equal to the minor part of the version space.

9.1. Asymptotic behaviour

Asymptotically for $q \to 1$ the answer is determined by the signs of the t_l's and the error probability is dominated by one-bitflip errors, so

$$\epsilon_{\text{Bayes}}(q) \to N_c \left(\frac{1}{2}\right)^{K-1} \int_0^\infty \mathrm{D}t \, H(\gamma t) \tag{68}$$

$$\to n_c \frac{1}{\pi} \sqrt{1-q}. \tag{69}$$

Comparing with the ϵ for Gibbs learning (34), called ϵ_{Gibbs}, we find that asymptotically for $q \to 1$

$$\epsilon_{\text{Gibbs}} = \epsilon_{\text{Bayes}} \sqrt{2}. \tag{70}$$

This relation is already known for the simple perceptron [21] and for the clock model [18]. It follows that for $\alpha \to \infty$

$$\epsilon_{\text{Bayes}}(\alpha) \to 0.4417\ldots\left(\frac{1}{\alpha}\right) \tag{71}$$

which is again independent of the architecture.

In the opposite limit (for small q) there is no change in the phase-transition behaviour compared to Gibbs learning: α_c and the order of the phase transition are unchanged. The reason is that both Bayes and Gibbs learning need non-vanishing q in the version space for non-trivial generalization, and q is the same in both cases since q is the typical overlap of two members of the version space.

9.2. An approximation and the central-point network

To obtain an upper limit for ϵ_{Bayes} we consider the following approximation. Instead of taking the minimum of the two terms in (67) we take the first term $(\text{Tr}\Delta^1\cdots)$ if $B(\{\text{sgn}(-t_l)\}) = -1$ and the second term $(\text{Tr}\Delta^{-1}\cdots)$ if $B(\{\text{sgn}(-t_l)\}) = 1$. Labelling the so defined approximation ϵ_{cpn}, then with

$$\int_0^\infty \text{D}t\, H(\gamma t) = \frac{1}{2\pi}\arccos\sqrt{q} \tag{72}$$

we obtain

$$\epsilon_{\text{cpn}}(q) = \left(\frac{1}{2}\right)^K \sum_{\sigma=\pm 1}\left[\operatorname*{Tr}_{\{\sigma_l^1\}}\Delta^{\sigma}(\{\sigma_l^1\})\operatorname*{Tr}_{\{\sigma_l^2\}}\Delta^{-\sigma}(\{\sigma_l^2\})\prod_l\left(1-\frac{1}{\pi}\arccos(\sigma_l^1\sigma_l^2\sqrt{q})\right)\right] \tag{73}$$

fulfilling the inequality

$$\epsilon_{\text{Bayes}} \leqslant \epsilon_{\text{cpn}}. \tag{74}$$

Comparing with (30) we see that

$$\epsilon_{\text{cpn}}(q) = \epsilon_{\text{Gibbs}}(\sqrt{q}). \tag{75}$$

Inserting this in the asymptotic form of ϵ_{Gibbs} for $q \to 1$ (34) we obtain

$$\begin{aligned}\epsilon_{\text{cpn}}(q) &\overset{q\to 1}{\to} n_c\frac{1}{\pi}\sqrt{2}\sqrt{1-\sqrt{q}} \\ &\to n_c\frac{1}{\pi}\sqrt{1-q}.\end{aligned} \tag{76}$$

This means that ϵ_{cpn} asymptotically reaches the Bayes generalization ability.

From the definition of ϵ_{cpn} we can directly construct a network which has a generalization error given by ϵ_{cpn}: considering all networks inside the version space, we take the average over the couplings of the subperceptrons for every hidden node:

$$\boldsymbol{w}_l^{\text{cpn}} = \frac{1}{C}\sum_j \boldsymbol{w}_l^j \tag{77}$$

where j counts all members of the version space and C takes care of the normalization $|\boldsymbol{w}_l^{\text{cpn}}|^2 = M$. The network with these central-point subperceptrons $\boldsymbol{w}_l^{\text{cpn}}$ in the first layer is called the central-point network. Analogous to the case of the simple perceptron, the overlap of this network with the teacher is $\sqrt{q}$, so the generalization error is given by ϵ_{cpn}.

9.3. The Bayes generalization of the parity-machine

We know that for the simple perceptron this central-point network has, in fact, the generalization ability of the Bayes algorithm for all α [18, 22]. We now ask whether there are other architectures with this characteristic. In other words, under which conditions does $\epsilon_{\text{cpn}} = \epsilon_{\text{Bayes}}$? This is simply the case when the final output $B(\{\text{sgn}(-t_l)\})$ of the central-point network equals the answer of Bayes algorithm for all patterns. We consider a point in the K-dimensional space of the common part $-t_l\sqrt{q}$, see (26), of the local fields corresponding to a random pattern. The central-point network gives the answer belonging to this point only. The Bayes algorithm gives an answer according to the majority vote of the members of the version space. Due to the distribution of the members in the version space the local fields are distributed spherically around the point given by the $-t_l\sqrt{q}$'s. So the Bayes algorithm considers a sphere around this point and considers whether the section of this sphere with the $+1$-area is larger than that with the -1-area or *vice versa*. (More accurately, it considers a lot of Gaussian-weighted spherical surfaces.) These strategies lead to the same result in two cases:

(i) in the limit $q \to 1$ for all networks, since the spheres shrink to their central point;

(ii) for the whole α-space, if and only if the network is a parity-machine:

$$\epsilon_{\text{cpn}}(\alpha) = \epsilon_{\text{Bayes}}(\alpha)\ \forall\alpha \ \Leftrightarrow\ B = \text{parity-machine}. \tag{78}$$

To see the special role of the parity-machine we consider the K-dimensional space of the local fields of the hidden nodes. The areas leading to final output one shall be shaded grey, the other regimes are white (as in figure 3). We now call the point given by the common part $-t_l\sqrt{q}$ of the local fields due to a test pattern the *central point* (belonging to this pattern). The colour of this point is the colour of the regime it lies in. The majority (K-dimensional) volume of a sphere around this point may be grey or white.

The point is that for the parity-machine the colour of the majority volume is always the colour of the central point, whereas this is not, in general, the case for all other decision functions. This means that the answer of the central-point network equals the Bayes prediction, in general, for the parity-machine but not for all others.

But why does the colour of the central point equal the colour of the majority volume just for the parity-machine? For $K = 2$, where only the parity- and the AND-machine exist, this can easily be seen from figure 3.

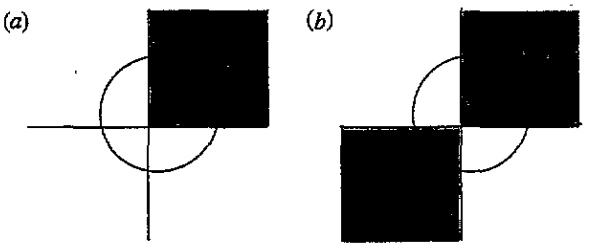

Figure 3. The plane of the common local field $-t_l\sqrt{q}$, see (26), for (*a*) the AND-machine and (*b*) the parity-machine is shown. The regimes with answer one are shaded. For the AND-machine the major volume of the sphere is in the white area, whereas the central point is in the grey area and, hence, for the AND-machine the answers of Bayes algorithm and of the central-point network are not the same in this example. For the parity-machine both the area of the central point and the majority area are grey, which is obviously so for all cases. This means that the Bayes algorithm and the central-point network always give the same answer for the parity-machine.

To extend the argument to higher-dimensions one uses induction from K to $K + 1$ to

show the statement for the parity-machine. One sees that this works because the space of the local fields is completely chequered (no neighboured areas have the same colour, see figure 3 for $K = 2$) in the case of the parity machine. For the other decision functions one considers that by fixing $K - 2$ values of the hidden nodes one has a two-dimensional AND, parity or constant function corresponding to the two free variables (since B is permutation symmetric, a ruler function can be excluded). It is easy to see that there exists at least one combination of the $K - 2$ values of the fixed parameters leading to an AND-machine due to the two free variables (again this holds only for permutation-symmetric B). Now one can use the $K = 2$ case by choosing a pattern with high stability in the fixed hidden-node values.

10. Conclusions

We investigated the generalization ability of tree architecture neural networks with fixed boolean decision functions B. Concentrating on learnable rules with a permutation symmetric function B we found for small α different behaviour concerning the occurrence of a phase transition.

We found certain simple characteristic values of B determining this behaviour or at least giving hints, namely (i) N_c, which is the border-regime size between the local-field volumes mapped to $+1$ and -1, (ii) the correlation moments a_m^σ of the class of IRs mapped to the final output σ, and (iii) the number n of the first non-vanishing correlation moment.

If $n \geqslant 3$, a phase transition of first order always occurs. If $n = 2$, a phase transition of second order occurs at $\alpha_c = \bar{\alpha}_c$ given by (63), except the system already undergoes a phase transition of first order at lower α_c. For $n = 1$, non-trivial generalization ability is reached at once from $\alpha > 0$ onwards. This means that one needs a preferred sign in the IRs mapped to certain σ to obtain non-trivial generalization for small α. Nevertheless, if the slope of the generalization error $\epsilon(\alpha)$ is small at $\alpha = 0$, a phase transition of first order is to be expected.

Heuristically we saw that the value of N_c is correlated to the occurrence of a first-order phase transition, so one can give the roughly fulfilled rule: only above a specific N_c does a machine have a phase transition of first order and the higher the value of N_c, the higher the value of α_c. Nevertheless the 'smooth' phase transition of second order can also occur for low N_c.

For $\alpha \to \infty$ the behaviour of $\epsilon(\alpha)$ is independent of the architecture. The phase-transition point α_c^{parity} of the parity-machine seems to be an upper limit for the occurrence of a phase transition and the asymptotic regime starts roughly at this α_c^{parity} (for a given number K of hidden nodes).

Considering the Bayes algorithm we found that the so called central-point network reaches Bayes generalization ability for every architecture asymptotically for $\alpha \to \infty$, whereas the parity-machine is as the only architecture type to have this characteristic over the whole α-space.

Acknowledgments

I thank U Krey, J Winkel, G Pöppel and F Gerl for helpful discussions and careful reading of the manuscript, and the HLRZ Jülich for computing time.

References

[1] Gardner E 1987 Maximum storage capacity in neural networks *Europhys. Lett.* **4** 481–5

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

[3] Schwarze H 1993 Learning a rule in a multilayer neural network *J. Phys. A: Math. Gen.* **26** 5781–94

[4] Schwarze H and Hertz J 1992 Generalization in a large committee machine *Europhys. Lett.* **20** 375–80

[5] Schwarze H and Hertz J 1992 Learning from examples in a fully connected committee machine *J. Phys. A: Math. Gen.* **26** 4919–36

[6] Kang K, Kwon C and Park Y 1993 Generalization in a two-layer neural network *Phys. Rev.* E **48** 4805–9

[7] Opper M 1994 Learning and generalization in a two-layer neural network: the role of the Vapnik–Chervonenkis dimension *Phys. Rev. Lett.* **72** 2113–6

[8] Mato G and Parga N 1992 Generalization properties of multilayered neural networks *J. Phys. A: Math. Gen.* **25** 5047–54

[9] Griniasty M and Grossman T 1992 Two-layer perceptron at saturation *Phys. Rev.* A **45** 8924–37

[10] Priel A, Blatt M, Grossman T, Domany E and Kanter I 1994 Computational capabilities of restricted two-layered perceptrons *Phys. Rev.* E **50** 577–95

[11] Barkai E, Hansel D and Kanter I 1990 Statistical mechanics of a multilayered neural network *Phys. Rev. Lett.* **65** 2312

[12] Barkai N, Hansel D and Sompolinsky H 1992 Broken symmetries in multilayered perceptrons *Phys. Rev.* A **45** 4146

[13] Engel A, Kohler H M, Tschepke F, Vollmayr H and Zippelius A 1992 Storage capacity and learning algorithms for two-layer neural networks *Phys. Rev.* A **45** 7590

[14] Gerl F and Krey U 1995 Replica symmetry breaking and the Kuhn–Tucker cavity method in simple and multilayer perceptrons *Preprint*

[15] Hansel D, Mato G and Meunier C 1992 Memorization without generalization in a multilayered neural network *Europhys. Lett.* **20** 471–6

[16] Engel A and Reimers L 1994 Reliability of replica symmetry for the generalization problem of a toy multilayer neural network *Europhys. Lett.* **28** 531–6

[17] Opper M and Kinzel W 1995 Statistical mechanics of generalization *Physics of Neural Networks* ed J L van Hemmen, E Domany and K Schulten (Berlin: Springer) to appear

[18] Schottky B and Gerl F and Krey U 1994 Generalization ability and information gain of clock-model perceptrons *Z. Phys.* B **96** 279–80

[19] Gerl F and Krey U 1995 A Kuhn–Tucker cavity method and the generalization ability of Potts-model perceptrons *Preprint*

[20] Watkin T L H, Rau A and Biehl M 1993 The statistical mechanics of learning a rule *Rev. Mod. Phys.* **65** 499

[21] Opper M and Haussler D 1991 Generalization performance of Bayes optimal classification algorithm for learning a perceptron *Phys. Rev. Lett.* **66** 2677–80

[22] Watkin T L H 1993 Optimal learning with a neural network *Europhys. Lett.* **21** 871–6