Proceedings Intern. AMSE Conference Neural Networks, San Diego (USA), May 29-31, 1991, Vol. 1, pp. 151-162

GENERALIZATION IN FEEDFORWARD NEURAL AND BOOLEAN NETWORKS

C. Van den Broeck*
Dept. Chem.

R. Kawai
Darpa Inst. Pure Appl. Phys. Sci.

0340 UCSD, La Jolla, CA 92093

Abstract.

An amazing feature of feedforward neural and Boolean networks is their ability to generalize. We present a rigorous lower bound for the probability for generalization. This bound can be calculated on the basis of the probability (or entropy) landscape, that characterizes the network prior to teaching taking place. The predicted learning curves agree extremely well with the results obtained using various teaching procedures in both neural and Boolean networks. Large fluctuations in the generalization ability are observed, indicating that the average or worst case performance may not be very representative.

Introduction.

We will study the following question: a feedforward network is trained to correctly classify a (non-exhaustive) set of teaching examples; what is the probability that the network will correctly classify new input patterns? Such an ability to generalize has been demonstrated quite convincingly in several neural [1] and Boolean networks [2-3]. At first, it may seem surprising that a network select one classification scheme, compatible with the teaching examples, rather then another compatible choice. The explanation is quite simple: the specific architecture and building blocks of the network restrict the number of classification schemes that can be implemented or establishes a hierarchy of hypotheses such that unlikely hypotheses are implemented by a comparatively small fraction of network configurations. For example, a perceptron with n binary-valued connection strengths has only 2^n different network configurations, and can only implement that many out of the total number of 2^{2^n} possible binary input-output

*Permanent address: Dept. Phys., L.U.C., 3590 Diepenbeek, Belgium.

tables. A richer structure was found for a feedforward Boolean net [3]: the various input-output tables are implemented with a priori probabilities that form a self-similar hierarchy. As teaching proceeds, the incompatible hypotheses are eliminated, and the probability landscape is redrawn in favor of the hypotheses that are similar to the classification scheme from which the teaching examples were drawn.

Previous theoretical approaches of generalization have established bounds on the generalization performance, based on a worst case analysis [4], or have calculated the statistical properties of an ensemble of networks [5]. Although these results are valuable, we believe that the interesting properties of the networks do not reside in the behavior of the worst-case scenario or the statistical average. On the contrary, a sizable number of specific input-output tables or hypotheses can be learned rather easily precisely at the "expense" of the mediocre average performance.

In previous work [3], we derived a rigorous lower bound for the generalization performance of a specific hypothesis, given in terms of the probability landscape prior to teaching. We review the derivation of this bound in section 2, and compare it to the results obtained using a similar approach that was formulated independently [6]. In section 3, we show that this lower bound agrees well with the results obtained using various teaching procedures in neural and Boolean networks.

For simplicity, we have restricted ourselves to networks with n binary input signals ± 1, and with a single binary output. This corresponds to a simple yes/no classification of the 2^n input patterns, or in other words, to the implementation of a n-variable Boolean function. Note that there are 2^{2^n} such input-output tables or Boolean functions (e.g. 65536 for n=4), each of which can be specified by the value of a running index i, i=1, 2^{2^n}.

2. The a priori probability landscape and Zipf's law.

The a priori probability p_i is defined as the probability that a randomly selected network configuration will implement the input-output table i. p_i can be obtained, either through exhaustive search in smaller networks, or through Monte Carlo analysis in larger networks.

For example, in Fig. 1a, we have represented the "entropy landscape", log p_i versus i, for a n=5 Boolean network, obtained by generating 10^7 randomly chosen network configurations. Such a network consists of ordinary Boolean gates, such as a XOR, AND, OR, ..., linked together in a feedforward but otherwise random manner (for more details, see [2-3]). Clearly, the landscape displays a hierarchical, self-similar structure. This is confirmed by the log-log plot of p_i versus rank r, with ranking according to decreasing values of p_i (i.e. p_i is ranked r if it is the r^{th} largest probability), cf. Fig. 1b.

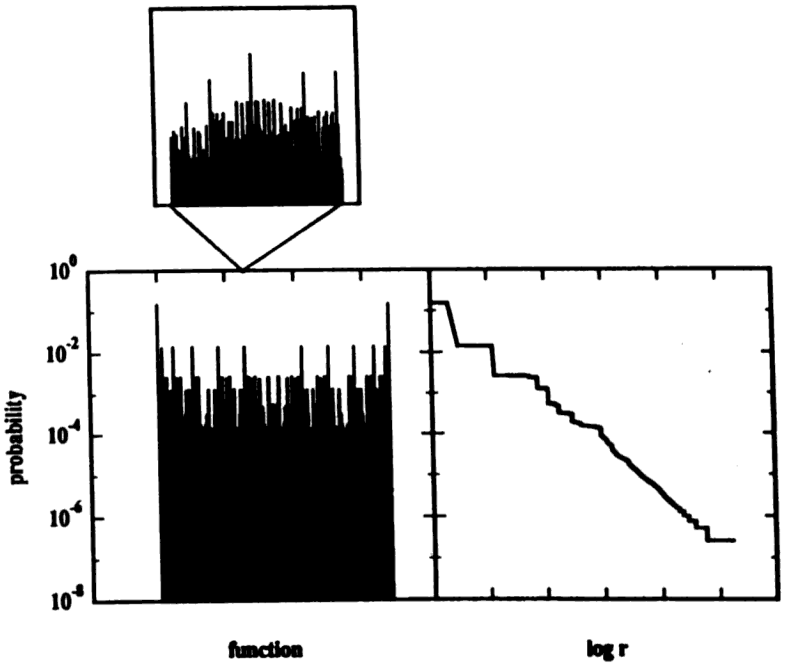

Fig. 1a
The a priori probability p_i versus i in a 5-input Boolean network.

Fig. 1b
Log-log plot of p_i versus rank r of table i.

The observed scaling behavior $p \sim 1/r^{\gamma}$ (with $\gamma \approx 1.3$) is similar to that encountered when the words of natural language are ranked in order of decreasing frequency of appearance (Zipf's law [7]). For comparison, we show the analogous results obtained for a 5-input perceptron with real-valued connection strengths J_i (chosen at random on the unit sphere $|\vec{J}| = 1$), in Fig. 2a and Fig. 2b respectively. We again observe scaling behavior. We finally note that the landscape is rather trivial in the case of a perceptron with binary connection strengths $J_i = \pm 1$: either $p_i = 1$ if table i can be implemented (e.g. the majority problem), else $p_i = 0$.

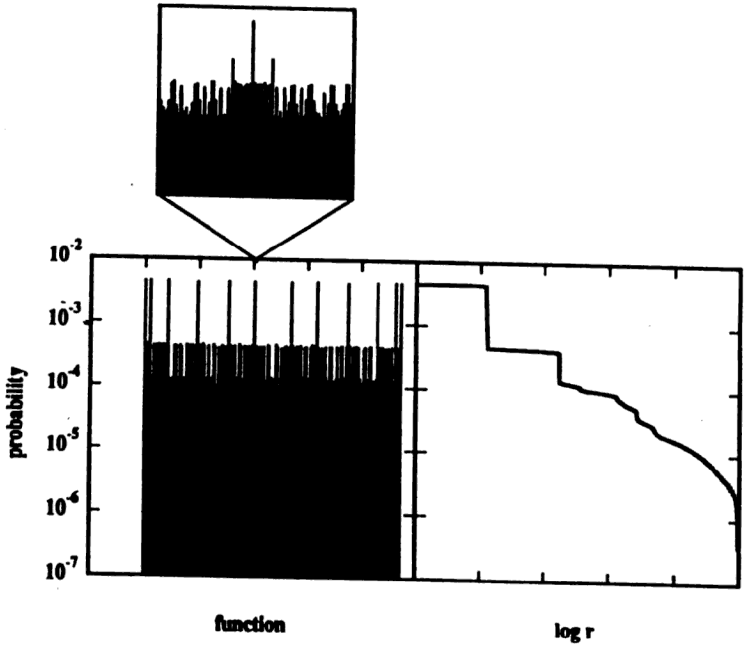

Fig. 2a
The a priori probability p_i versus i in a 5-input perceptron.

Fig. 2b
Log-log plot of p_i versus the rank r of table i.

3. Lower bound for generalization.

We now derive a lower bound for the generalization probability $P_i(L)$ that the network will implement the input-output table i without a single error, after being trained to correctly classify L teaching examples from this table. Typically, this probability depends on the choice of the teaching examples, and we define $P_i(L)$ as the ensemble average over all these choices. Clearly, a network trained on L examples of table i, with $L < 2^n$, can still implement a table $j \neq i$, provided this classification table is compatible with the teaching examples. If the teaching procedure is equivalent to a random search in the space of compatible configurations, the probability to choose such a classification is p_j, and $P_i(L)$ is given by:

$$P_i(L) = \left\langle \frac{p_i}{\sum_{j \text{ is compatible with the L teaching examples}} p_j} \right\rangle_{\text{average over the choice of the L teaching examples}} \qquad (1)$$

The average over the choice of the L teaching examples is very difficult to perform at this stage, but using Jensen's inequality $\langle 1/x \rangle \geq 1/\langle x \rangle$, for any random variable x taking positive values only, one obtains the following rigorous lower bound:

$$P_i(L) \geq \frac{p_i}{\left\langle \sum_{j \text{ is compatible with the L teaching examples}} p_j \right\rangle_{\text{average over the choice of the L teaching examples}}} \qquad (2)$$

For the further discussion, it is convenient to define the Hamming distance d between two Boolean functions as the number of different classifications in their respective output tables, $0 \leq d \leq 2^n$. The sum appearing in the denominator of the r.h.s. of Eq. (2) can then be rewritten as follows:

$$< \sum_{\substack{j \text{ is compatible with} \\ \text{the L teaching examples}} } p_j > \quad \text{average over the choice of the L teaching examples}$$

$$= \sum_{d=0}^{2^n - L} < \sum_{\substack{j \text{ is compatible with} \\ \text{the L teaching examples and} \\ \text{is at Hamming distance d from i}} } p_j > \quad \text{average over the choice of the L teaching examples} \qquad (3)$$

The point is now that the averages appearing in the r.h.s. of (3) can be calculated exactly in terms of the a priori probabilities $\{p_i\}$:

$$< \sum_{\substack{j \text{ is compatible with} \\ \text{the L teaching examples and} \\ \text{is at Hamming distance d from i}} } p_j > \quad \text{average over the choice of the L teaching examples} \quad = \frac{\binom{2^n - L}{d}}{\binom{2^n}{d}} p_i^{(d)} \qquad (4)$$

where we introduced the neighborhood function (note that $p_i^{(0)} = p_i$):

$$p_i^{(d)} = \sum_{\substack{j \text{ is at Hamming} \\ \text{distance d from i}} } p_j \qquad (5)$$

We conclude that:

$$P_i(L) \geq \frac{p_i}{\sum_{d=0}^{2^n-L} \frac{\binom{2^n - L}{d}}{\binom{2^n}{d}} p_i^{(d)}} \qquad (6)$$

A rigorous lower bound to $P_i(L)$ is thus obtained in terms of the a priori probability landscape $\{p_i\}$ of the untrained network. Other quantities

can be calculated in a similar way, but the average over the choice of the teaching examples can only be done approximately. For example, the probability $G_i(L)$ that a new example will be classified correctly according to table i, after L teaching examples from this table have been learned, is given by the following approximate result:

$$G_i(L) \approx \frac{\sum_{d=0}^{2^n-L} (2^n - L - d) \frac{\binom{2^n - L}{d}}{\binom{2^n}{d}} p_i^{(d)}}{\sum_{d=0}^{2^n-L} (2^n - L) \frac{\binom{2^n - L}{d}}{\binom{2^n}{d}} p_i^{(d)}} \qquad (7)$$

The combinatorial factors, appearing in Eqs. (6) and (7) can be simplified considerably, if we assume that the learning transition occurs in the region $L \ll 2^n$, and that the dominant contribution to the sum comes from values $d \gg L$ and $d \ll 2^n - L$. In this case, we get the following estimates:

$$P_i(L) \approx \frac{p_i}{\sum_{d=0}^{2^n-L} \left(1 - \frac{d}{2^n}\right)^L p_i^{(d)}} \qquad (8)$$

$$G_i(L) \approx \frac{\sum_{d=0}^{2^n-L} \left(1 - \frac{d}{2^n}\right)^{L+1} p_i^{(d)}}{\sum_{d=0}^{2^n-L} \left(1 - \frac{d}{2^n}\right)^L p_i^{(d)}} \qquad (9)$$

Eq. (9) is identical to the result obtained by Schwartz et al. [6]. In most applications reported in the following section, the differences between the results obtained from Eqs. (6) and (7) and from Eqs. (8) and (9) respectively, are small. For lack of space, we only report the results obtained from Eq. (6).

3. Applications.

In Fig. 3a, we show the learning curve $P_i(L)$ for the parity problem in a 4-input feedforward Boolean network. The full line represents the lower bound obtained from Eq. (6), the circles correspond to the results of a random search teaching procedure, and the triangles are the results of a simulated annealing teaching procedure (for more details, see [3]). The agreement between the lower bound and the results obtained through teaching is rather good. The error bars indicate the amplitude of the fluctuations around the ensemble average $P_i(L)$ from one set of teaching examples to another.

Similar results are obtained for the case of neural networks. In Fig. 3b, we have plotted the learning curves obtained from Eq. (6) (full line) and from a random search algorithm (circles) for the majority problem in a 15 input perceptron with binary connection strengths $J_i = \pm 1$. In Fig. 3c we give the corresponding results for the majority problem in a 5 input perceptron with continuous weights. We also included the learning curve found by applying the perceptron teaching algorithm (cf. triangles). Finally, the learning curves for the contiguity problem in a feedforward network with a hidden layer (for more details, see reference [6]) are reproduced in Fig. 3d. The results for Figs. 3b and 3d were obtained through exhaustive search of all network configurations on the Connection Machine, and are thus exact.

In all the above cases the lower bound for $P_i(L)$ given in Eq. (6) gives a rather accurate picture of the actual learning features of the network. However, we repeat that this result refers to an average over the choice of the teaching examples, and the learning curve obtained for the "worst choice" of these examples may actually lie below this lower bound. This for instance happens when the generalization probability strongly differs from one choice of teaching examples to another one.

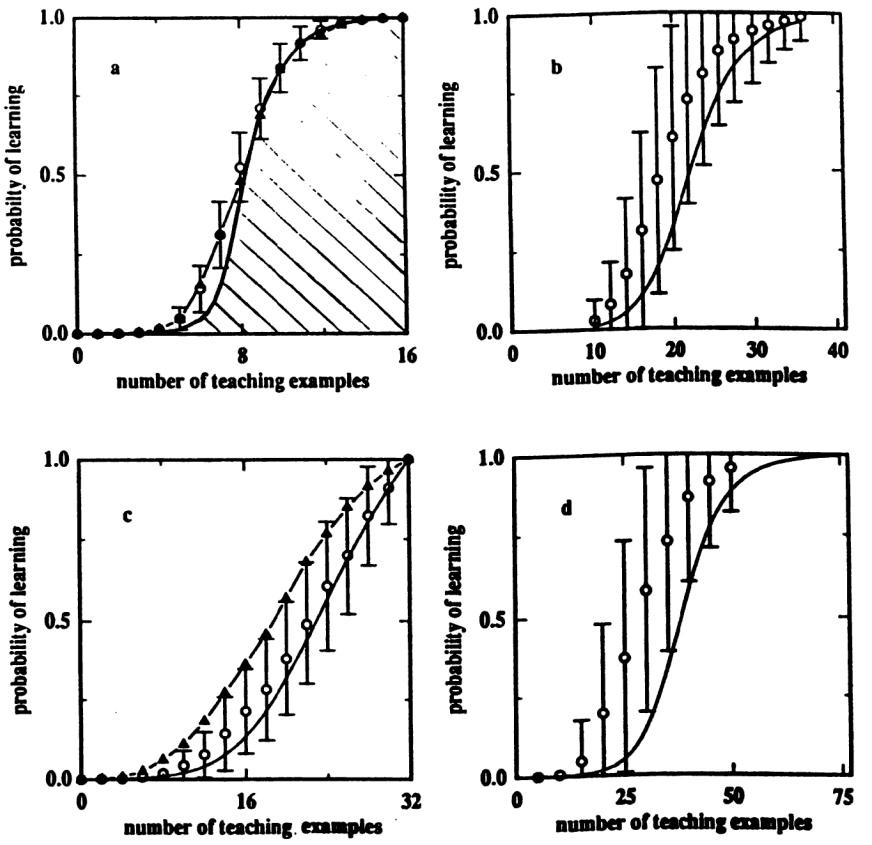

Fig. 3

The generalization probability $P_i(L)$ in function of the number of teaching examples, obtained from Eq. (6) (full line) and by applying random search (circles) or other teaching procedures (triangles) for the Boolean network (Fig. 4a), the perceptron with binary (Fig. 4b) and continuous connection strengths (Fig. 4c), and a network with a hidden layer (Fig. 4d).

Such large fluctuations are observed in the region where the "Eureca" learning transition takes place, a phenomenon reminiscent of the critical fluctuations that are observed in the vicinity of phase transitions.

4. Discussion.

Generalization implies that one disregards some possibilities in favor of other more "reasonable" hypotheses. The perceptron with binary connection strengths can generalize because a large number of input-output relations are completely "disregarded", i.e. they cannot be implemented at all. A more interesting situation arises in more complicated networks: a scaling hierarchy of hypotheses with decreasing a priori probability p_i is established. By eliminating incompatible hypotheses with large a priori probability, the network can select a hypotheses with comparatively small a priori probability p_i. To quantify these ideas, we have derived a rigorous lower bound for the generalization probability $P_i(L)$. This bound is valid if one assumes that the teaching procedure is equivalent to a random search in the space of compatible network configurations. The existence of a fractal-like landscape $\{p_i\}$ leads to learning curves $P_i(L)$ that can differ strongly from one hypotheses i to another, with the average generalization curve being of a rather mediocre value. To illustrate this point, we have represented in Fig. 4 the "learning capacity" of all 2^{16} input-output tables i for a 4-input Boolean network, in function of the a priori probability p_i. This "learning capacity", taking values in the interval [0,1], is defined as the surface below the learning curve $P_i(L)$, plotted in function of the fraction of teaching $L/2^n$ (cf. cross-hatched region in Fig. 3a). As is clear from Fig. 4, a wide range of learning capacities are obtained, substantiating our claim that the average learning curve, apart from being of very mediocre value, does not reflect the typical behavior of the network. Finally, we note that a large a priori probability p_i does not necessarily imply a large learning capacity. The teaching eliminates incompatible tables, and more likely so those that are at a large Hamming distance from the table under consideration. As a result, good generalization is observed for the tables that correspond

to a local maximum of the a priori probability p_i, local being defined with respect to the Hamming distance (for more details, see [3]).

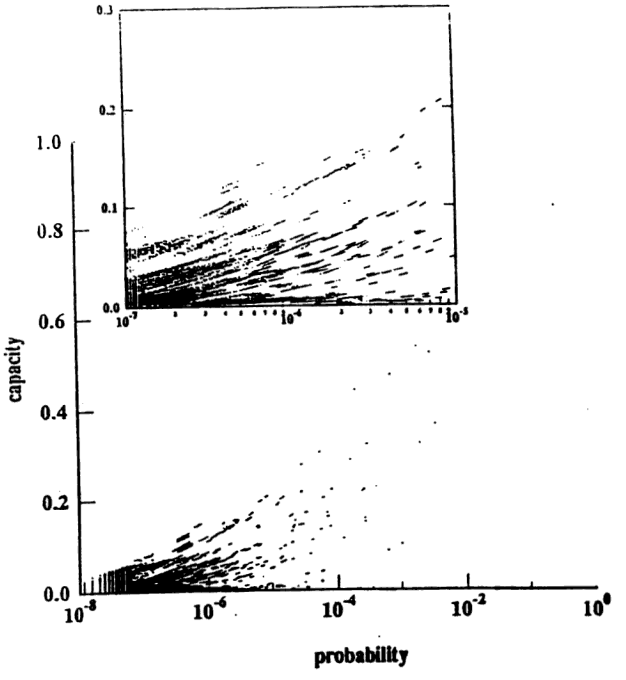

Fig. 4

We evaluated numerically the capacity of all the 2^{16} tables i in a 4-input Boolean network. A dot in this log-log plot corresponds to a capacity plotted versus the corresponding p_i.

Acknowledgments.

We would like to thank Prof. K. Lindenberg and Prof. J. Weare for their support. C. Van den Broeck acknowledges financial support from DOE grant DE-FG03-86ER13606, from the Program on Inter-University Attraction Poles, Prime Minister's Office, Belgian Government and from the N.F.W.O. Belgium. R. Kawai thanks Dr. M. Marron for providing him with computer time on the Connection Machine through contract N00014-87-K-0675.

References.

1. Some recent references are:
   Neural Computing Architectures, Ed. I. Aleksander (M.I.T.-press, Cambridge, 1989),
   Advances in Neural Information Processing Systems 1, Ed. D.S. Touretzky (Morgan Kaufmann Publ., San Mateo, CA, 1989),
   Lectures in the Sciences of Complexity, Ed. D. Stein (Addison-Wesley Publ. Comp., Redwood City, CA, 1989),
   Advances in Neural Information Processing Systems 2, Ed. D.S. Touretzky (Morgan Kaufmann Publ., San Mateo, CA, 1990).
2. S. Patarnello and P. Carnevali, Europhys. Lett.4, 503 (1987).
   P. Carnevali and S. Patarnello, Europhys. Lett. 4, 1199 (1987).
3. C. Van den Broeck, Entropy and Learning, to be published in the proceedings of the NATO workshop: Self-Organization, Emerging Properties and Learning, Austin, March 12-14,1990;
   C. Van den Broeck and R. Kawai, Phys. Rev. A42, 6210 (1990).
4. E. B. Baum and D. Haussler, Neural Computation 1, 151 (1989);
   T.M. Mitchell, Artificial Intelligence 18, 203 (1982).
5. E. Gardner and B. Derrida, J. Phys. A22, 1983 (1989).
6. D. B. Schwartz, V.K. Samalam, S.A. Solla and J.S. Denker, Neural Computation 2, 374 (1990).
7. J. R. Pierce, An Introduction to Information Theory (Dover, New York, 1980).