Information theoretical approach to the storage capacity of neural networks with binary weights

Hiroki Suyari* and Ikuo Matsuba†
Department of Information and Image Sciences, Faculty of Engineering, Chiba University 1-33, Yayoi-cho, Inage-ku, Chiba-shi, Chiba 263-8522 Japan

(Received 5 January 1999; revised manuscript received 17 May 1999)

The storage capacity of the perceptron with binary weights $w_i \in \{0,1\}$ is derived by introducing the minimum distance d between input patterns. The approach presented in this paper is based on some results in the information theory, and the obtained storage capacity 0.585 is in good agreement with the well-known value 0.59 by the replica method in statistical physics. A strength of the present information theoretical approach is that it provides an easier and more intuitive understanding for the storage capacity than the replica method, which is believed to be more reliable and informative than the Vapnik-Chervonenkis procedure.
[S1063-651X(99)12109-3]

PACS number(s): 87.10.+e

The storage capacities of artificial neural networks (ANN) has been investigated by means of various concepts in mathematical physics [1–3]. The replica method in statistical physics [4] especially has succeeded in finding some storage capacities concretely [5–7]. This replica method now becomes the most useful tool to analyze the storage capacity and generalization ability of ANN [1–3,8–10]. ANN is obviously one of the information transmissions, indicating that it is natural to derive and understand these storage capacities of ANN in the framework of information sciences. There already exists some information theoretical approaches to obtain the storage capacities of ANN [11,12], but most of them do not succeed in obtaining them concretely. In the present paper, the storage capacity of the perceptron with binary weights $w_i \in \{0,1\}$ is derived by introducing the minimum distance d between input patterns. We obtained $\alpha_c = 0.585$ that should be compared with the well-known value $\alpha_c = 0.59$ [9] by the replica theory, which means that the ideas presented in this paper can be applied to the computation of these storage capacities without the usual replica method.

The perceptron is usually formulated by

$$y = \mathrm{sgn}\left(\sum_{i=1}^{n} w_i x_i - \theta \right) \tag{1}$$

with weights $w_i (i = 1, \ldots, n)$ and a threshold value θ for each input pattern $(x_1, \ldots, x_n)^t$ and two-valued output $y \in \{-1, +1\}$ [1,2].

Our main ideas are based on introducing the minimum distance d between input patterns. The introduction of this parameter d helps in understanding the basic ingredient underlying the storage capacity as stated below. In order to explain our ideas, we briefly review the definitions of the capacities such as Vapnik-Chervonenkis (VC) capacity and the storage capacity [13].

*Electronic address: suyari@ics.tj.chiba-u.ac.jp
†Electronic address: matsuba@ics.tj.chiba-u.ac.jp

For the perceptron given by Eq. (1), each input pattern can be described as a n-dimensional column vector: $|x^{(k)}\rangle \equiv (x_1^{(k)}, \ldots, x_n^{(k)})^t \in \{0,1\}^n$ where the index k denotes the kth pattern. If the neural network can learn the kth input pattern $|x^{(k)}\rangle$, then there exists a row vector $\langle w^{(k)}| \equiv (w_1^{(k)}, \ldots w_n^{(k)})$ for the kth input pattern $|x^{(k)}\rangle$ such that the input-output relation is written as $y^{(k)} = \mathrm{sgn}(\langle w^{(k)}|x^{(k)}\rangle - \theta)$ where $\langle w^{(k)}|x^{(k)}\rangle \equiv \Sigma_{i=1}^{n} w_i^{(k)} x_i^{(k)}$ is an inner product.

Let $S^{(n,p)} \equiv \{|x^{(1)}\rangle, \ldots, |x^{(p)}\rangle\} \subset \{0,1\}^n$ be any set of p different input patterns. For simplicity, we write down S for $S^{(n,p)}$, if not necessary. The number $\Delta(S)$ of different output vectors $(y^{(1)}, \ldots, y^{(p)})$ can be determined by Eq. (1) for any set S. The growth function $\Delta(p)$ and the typical growth function $\Delta^{typ}(p)$ are defined by

$$\Delta(p) \equiv \max_{|S|=p} \Delta(S), \quad \Delta^{typ}(p) \equiv \mathrm{mean}_{|S|=p} \Delta(S), \tag{2}$$

respectively, where $|S|$ represents the number of elements in a set S [13]. The Vapnik-Chervonenkis capacity α_{VC} [14] and the storage capacity α_c are, respectively, given by [13]

$$\alpha_{VC} \equiv \lim_{n\to\infty} \frac{p_{VC}}{n}, \quad \alpha_c \equiv \lim_{n\to\infty} \frac{p_c}{n}, \tag{3}$$

where p_{VC} is the Vapnik-Chervonenkis dimension [15,16]

$$p_{VC} \equiv \max\{p \in \mathrm{N} | \Delta(p) = 2^p\}, \tag{4}$$

and p_c is a solution to

$$\frac{\Delta^{typ}(p_c)}{2^{p_c}} = \frac{1}{2}. \tag{5}$$

There is extensive literature on the VC capacity in a variety of fields. [17–24]. Much of the previous works on the storage capacity has been discussed by means of the replica method [1–3,5–10].

In the above definitions of α_{VC} and α_c, it is important to emphasize that a set S does depend only on p. When we consider the capacity of neural networks, it is natural to take p as a parameter in order to solve the following problem: (a)

How many input patterns can neural networks learn? Unfortunately, little is known about the exact form of $\Delta(S)$ as a function of p, and it is thus difficult to find the distribution $\Delta(S)$ with respect to S.

Therefore, we propose the minimum distance d among input patterns S as a new parameter to characterize S. Instead of (a), we then address the following problem: (b) What is the minimum distance d among input patterns that can be learned by neural networks? For a neural network to learn the maximum number of patterns, there must exist a minimum distance d for which the network distinguishes patterns. That is, the capacity can be considered to be dominated by the distinguishable minimum distance. After all, the introduction of d can solve original problem (a). What is better, the information theory yields an exact formula of lower and upper bounds on the maximum number of different inputs with d.

Before introducing d, we should take the following into account: (i) What is the best representation of the input patterns for the neural networks? (ii) What is the best definition of the distance d among the input patterns? The former is the problem of coding of the information resources for the neural networks, which is irrelevant to the computation of its capacity. This coding depends on what and how the neural networks recognize, which can be also discussed in the information theory. The latter is concerned with the definition of a distance that is often seen in any mathematical books on topology as a map satisfying the three conditions, namely, separability, symmetry, and trigonometric inequality. The choice of the definition of a distance depends on what the neural networks can distinguish. In this paper we take the Hamming distance d as a typical distance among input patterns [25]. The Hamming distance $d(|x^{(k)}\rangle,|x^{(l)}\rangle)$ between any two input patterns $|x^{(k)}\rangle$ and $|x^{(l)}\rangle$ is defined by $d(|x^{(k)}\rangle,|x^{(l)}\rangle)\equiv\Sigma_{i=1}^{n}(x_i^{(k)}\oplus x_i^{(l)})$ where $x_i^{(k)}\oplus x_i^{(l)}=0(x_i^{(k)}=x_i^{(l)}),1(x_i^{(k)}\neq x_i^{(l)})$. Using this distance d, we can correspond $A(n,d)$ to $\Delta(p)$ in the follwing manner. $\Delta(p)$ is the maximum number of classifications of all S through p input patterns. On the other hand, $A(n,d)$ represents the maximum number of codewords in any binary input of length n and minimum distance d in the information theory [25]. $\Delta(p)$ can be characterized by the minimum distance d between p input patterns S. Thus, for a given p there must exist the minimum distance d to satisfy

$$A(n,d)=\Delta(p). \tag{6}$$

Equation (6) gives d as a function of p. However, Eq. (6) cannot be solved, because the exact formula of $\Delta(p)$ is unknown. On the other hand, the exact expression of the upper and lower bound of $\lim_{n\to\infty}\log_2 A(n,d)/n$ are already known as the famous formula [see Eq. (9)] in the information theory. Therefore, we introduce parameter d instead of the usual parameter p in the sense that we can apply $A(n,d)$ to the computation of the storage capacity instead of the usual $\Delta(p)$. After this consideration, we concentrate on computing the mean value $\Delta^{typ}(p)$, which is the average value of $A(n,d)$ with respect to $x=d/n$ under the condition $|S|=p$ in the rest of this paper.

The parameter p in the growth function $\Delta(p)$ is regarded as the length k of the information bits in a codeword with length n by the following considerations. In constructing an error-correcting code [25], any codeword consists of *information bits* and *check bits*, which represent the coded information resources and the redundancy for error correcting, respectively. In the information theory the codeword with length n consisting of information bits k and check bits $n-k$ is usually expressed by $[n,k]$ *code*.

Any $[n,k]$ code can represent 2^k different codes, whose number corresponds to $\Delta(p)$ given by Eq. (6), that is,

$$2^k=A(n,d)=\Delta(p). \tag{7}$$

The maximum number of information bits k can be taken as n, but in such a case $(k=n)$ it becomes quite difficult to distinguish these codewords (input patterns) because the distance among codewords is too short to distinguish them. The same situation occurs in the neural networks. Increasing the parameter p in $\Delta(p)$ makes it more difficult to distinguish p input patterns due to the short distance among them. This result has been known as the Sauer's lemma [15,26] in terms of $\Delta(p)$. Then, p_{VC} (the maximum number of input patterns) should be equal to k(length of information bits). Finding the information bits k is equivalent to determining the number of represented codewords. The ratio k/n is called *information ratio* in the information theory [25]. The larger k/n is, the more codewords they can represent. From Eqs. (6) and (7), k/n is written as

$$\frac{k}{n}=\frac{\log_2 A(n,d)}{n}=\frac{p_{VC}}{n}. \tag{8}$$

Note that p_{VC} can be expressed as a function of d, that is, $p_{VC}=\log_2 A(n,d)$ from the above Eq. (8).

The asymptotic bound on k/n for large n has been discussed in detail in the information theory [25]. The lower and upper bounds are known to be *the Gilbert-Varshmov bound* and *the McEliece-Rodemich-Rumesey-Welch bound*, respectively, given by

$$1-H_2\left(\frac{d}{n}\right)\leqslant\lim_{n\to\infty}\frac{\log_2 A(n,d)}{n}\leqslant B\left(\frac{d}{n}\right), \tag{9}$$

where $H_2(x)\equiv -x\log_2 x-(1-x)\log_2(1-x)$, $B(\delta)\equiv\min_{0<u\leqslant 1-2\delta}B(u,)$, $B(u,\delta)\equiv 1+h(u^2)-h(u^2+2\delta u+2\delta)$, and $h(x)\equiv H_2(\frac{1}{2}-\frac{1}{2}\sqrt{1-x})$. Figure 1 displays k/n $(=p_{VC}/n)$ as a function of d/n

Since the lower and upper bounds decrease monotonally with d/n as seen from Fig. 1, there must exist $\lambda(0\leqslant\lambda\leqslant 1)$ satisfying

$$\lim_{n\to\infty}\frac{p_{VC}}{n}=\lim_{n\to\infty}\frac{\log_2 A_n(x)}{n}=\lambda[1-H_2(x)]+(1-\lambda)B(x), \tag{10}$$

where $A_n(x)\equiv A(n,d)$ and $x\equiv d/n$, which we call a minimum distance ratio. Here we define $f_n(x)$ and $f(x)$:

$$f_n(x)\equiv\frac{\log_2 A_n(x)}{n}, \tag{11}$$

$$f(x)\equiv\lambda[1-H_2(x)]+(1-\lambda)B(x), \tag{12}$$

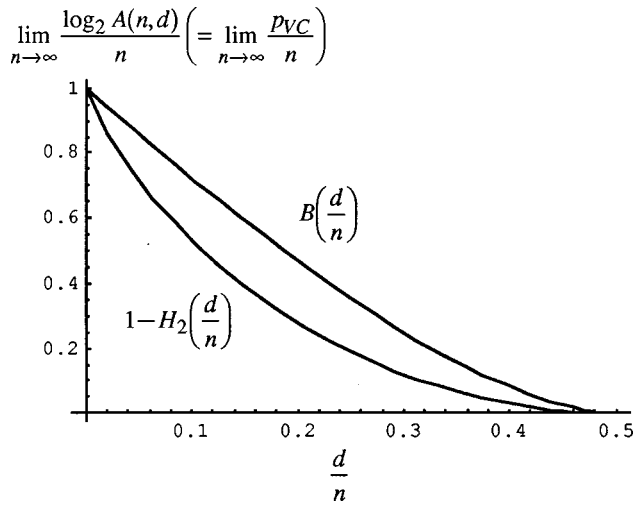

FIG. 1. Asymptotic bounds on $\lim_{n\to\infty} \log_2 A(n,d)/n$ $(=\lim_{n\to\infty} p_{VC}/n)$ as a function of d/n for $n\to\infty$.

respectively. Then from Eq. (10) we have

$$f(x)=\lim_{n\to\infty} f_n(x). \tag{13}$$

Substituting Eq. (7) into Eq. (11) yields

$$\Delta(p)=A_n(x)=2^{nf_n(x)}. \tag{14}$$

Let $\rho_n(x)$ be a probability density function. Since $A(n,d)[=A_n(x)]$ is the maximum number of codewords for a fixed d, we can write $A_n(x)$ in the form

$$\int_x^{0.5} \rho_n(x')dx'=\frac{A_n(x)}{C} \tag{15}$$

using a normalization C. It is clear that $C=A_n(0)=2^n$. Thus $\Delta^{typ}(p)$ can be described in terms of $\rho_n(x)$ as

$$\Delta^{typ}(p)=\int_0^{x_p} \rho_n(x)\Delta(S_x)dx, \tag{16}$$

where S_x is the set of input patterns S with a constant minimum distance ratio x. Since there always exists a maximum value of the minimum distance for a given p, we denote a maximum value of x by x_p. Instead of Eq. (5), consider the general case

$$\frac{\Delta^{typ}(p_r)}{2^{p_r}}=r, \tag{17}$$

where $0\leqslant r\leqslant 1$. For any $x\in[0,x_p]$ we have $\Delta(S_x)=2^p$. Equation (17) is thus written as

$$\int_0^{x_{p_r}} \rho_n(x)dx=r. \tag{18}$$

Equation (18) tells us the following interpretations for the capacity of neural networks by means of a distance d: For small $r(\cong 0)$, the distinguishable minimum distance ratio x_{p_r} is small $(x_{p_r}\cong 0)$ from Eq. (18), where from Eq. (10) and Fig. 1 $\lim_{n\to\infty} p_r/n$ is large $(\lim_{n\to\infty} p_r/n\cong 1)$. In this case neural networks can distinguish input patterns whose distance between them is small, and thus the capacity takes a large value. On the other hand, for large $r(\cong 1)$, x_{p_r} is large $(x_{p_r}\cong 0.5)$, where $\lim_{n\to\infty} p_r/n$ is small $(\lim_{n\to\infty} p_r/n\cong 0)$.

If we succeed in finding the concrete expression for $\int_0^{x_p}\rho_n(x)dx$ in the limit of large n, it is an easy task to obtain x_{p_r}, from which we have $\lim_{n\to\infty} p_r/n$ from Eq. (10).

To this end, we have

$$\int_0^x \rho_n(x')dx'=\frac{2^n-2^{nf_n(x)}}{2^n} \tag{19}$$

from Eqs. (14) and (15). The right-hand side of Eq. (19) is expanded to give

$$\int_0^x \rho_n(x')dx'=(2-2^{f_n(x)})\sum_{k=1}^{n} \frac{2^{(k-1)f_n(x)}}{2^k}. \tag{20}$$

Any $f_n(x)$ satisfies $0\leqslant f_n(x)\leqslant 1$. Substituting $f_n(x)=0$, $1-\varepsilon_n$, respectively, into the second factor of the right-hand side, it is straightforward to show that

$$(2-2^{f_n(x)})\sum_{k=1}^{n} 2^{-k}\leqslant \int_0^x \rho_n(x')dx'\leqslant(2-2^{f_n(x)})K_n, \tag{21}$$

where $\lim_{n\to\infty}\varepsilon_n=0$ and $K_n\equiv\Sigma_{k=1}^{n}2^{(\varepsilon_n-1-k\varepsilon_n)}$. Both Eqs. (20) and (21) hold for any x, so that we can assume the existence of K_n' such as

$$\int_0^x \rho_n(x')dx'=(2-2^{f_n(x)})K_n', \tag{22}$$

where $\Sigma_{k=1}^{n}2^{-k}\leqslant K_n'\leqslant K_n$. When $n\to\infty$, we get

$$\int_0^x \rho(x')dx'=(2-2^{f(x)})K_\infty', \tag{23}$$

where $K_\infty'\equiv\lim_{n\to\infty}K_n'$. Since the right-hand side of the above Eq. (23) is probability, $K_\infty'=1$. Thus we obtain

$$\int_0^x \rho(x')dx'=2-2^{f(x)}. \tag{24}$$

From Eqs. (18) and (24), for a given r we can derive x_{p_r} by only solving the equation,

$$2-2^{f(x_{p_r})}=r. \tag{25}$$

Using x_{p_r} by Eq. (25), we get

$$\lim_{n\to\infty}\frac{p_r}{n}=f(x_{p_r}). \tag{26}$$

For $r=1/2$ [see Eq. (5)] we have only to solve the following equation:

$$2-2^{f(x_{p_c})}=\tfrac{1}{2}. \tag{27}$$

For $\lambda=1,0$ we found $x_{p_c}=0.083,\ 0.153$, respectively. In both cases, $f(x_{p_c})=0.585$. Therefore, we can conclude

$$\alpha_c=\lim_{n\to\infty}\frac{p_c}{n}=f(x_{p_c})=0.585, \tag{28}$$

which agrees well with the well-known value 0.59 [9] obtained by the replica method for $w_i\in\{0,1\}$. In principle, it is possible to obtain $\Delta^{typ}(p)$ for simple multilayer networks. Moreover, using the present method, it is unnecessary to think whether replica symmetry breaking shows up.

In summary, these information theoretical derivations of the storage capacity are completely different from the usual replica method. The present approach is undoubtedly easier and more intuitive than the replica method. Moreover, the storage capacity for the case of $w_i\in\{-1,+1\}$ can be also computed by easy transformation of this method and its result is also in agreement with the result $\alpha_c=0.83$ obtained by the replica theory [27]. This derivation will be presented in our forthcoming paper [28].

[1] D.J. Amit, *Modeling Brain Function* (Cambridge University Press, Cambridge, 1989).

[2] J. Hertz, A. Krogh, and R.G. Palmer, *Introduction to the Theory of Neural Computation* (Addison-Wesley, Reading, PA 1991).

[3] E. Domany, J.L. van Hemmen, and K. Schulten, *Models of Neural Networks III* (Springer, New York, 1995), Chap. 5.

[4] T.L.H. Watkin, A. Rau, and M. Biehl, Rev. Mod. Phys. **65**, 499 (1993).

[5] E. Gardner, Europhys. Lett. **4**, 481 (1987).

[6] E. Gardner, J. Phys. A **21**, 271 (1988).

[7] E. Gardner and B. Derrida, J. Phys. A **21**, 271 (1988).

[8] W. Krauth and M. Opper, J. Phys. A **22**, L519 (1989).

[9] H. Gutfreund and D. Stein, J. Phys. A **23**, 2613 (1990).

[10] B. Derrida, R.B. Griffith, and A. Prügel-Bennett, J. Phys. A **24**, 4907 (1991).

[11] D. Haussler and A.R. Barron, in *Proceedings of the Third NEC Symposium on Comp. and Cogni.*, edited by E. Baum (SIAM, Philadelphia, PA, 1993).

[12] D. Haussler, M. Kearns, and R. Schapire, Machine Learning **14**, 83 (1994).

[13] S. Mertens and A. Engel, Phys. Rev. E **55**, 4478 (1997).

[14] M. Opper, Phys. Rev. E **51**, 3613 (1995).

[15] V. Vapnik and A. Chervonenkis, Theor. Probab. Appl. **16**, 264 (1971).

[16] V. Vapnik, *Estimation of Dependences Based on Empirical Data* (Springer-Verlag, Berlin, 1982).

[17] E.B. Baum and D. Haussler, Neural Comput. **1**, 151 (1989).

[18] A. Blumer, A. Ehrenfeucht, D. Haussler, and M.K. Warmuth, J. Assoc. Comput. Mach. **36**, 929 (1989).

[19] D. Cohn and G. Tesauro, Neural Comput. **4**, 249 (1992).

[20] P.L. Bartlett, Neural Comput. **5**, 371 (1993).

[21] W. Maass, Neural Comput. **6**, 877 (1994).

[22] S. Floyd and M. Warmuth, Machine Learning **21**, 269 (1995).

[23] A. Sakurai, Theor. Comput. Sci. **137**, 109 (1995).

[24] P. Koiran and E.D. Sontag, Discrete Appl. Math. **86**, 63 (1998).

[25] F.J. MacWilliams and N.J.A. Sloane, *The Theory of Error-Correcting Codes* (North-Holland, Amsterdam, 1977), Chap. 17.

[26] N. Sauer, J. Comb. Theory, Ser. A **13**, 145 (1972).

[27] W. Krauth and M. Mézard, J. Phys. (Paris), Colloq. **50**, 3057 (1989).

[28] H. Suyari and I. Matsuba (unpublished).