

Lower Bound for Connectivity in Local-Learning Neural Networks*

YASER S. ABU-MOSTAFA

Departments of Electrical Engineering and Computer Science, California Institute of Technology, Pasadena, California 91125

Received April, 1987

How does the connectivity of a neural network (number of synapses per neuron) relate to the complexity of the problems it can handle? Switching theory would suggest no relation at all, since all Boolean functions can be implemented using a circuit with very low connectivity (e.g., using two-input NAND gates). However, for a network that learns a problem from examples using a local learning rule, we prove that the entropy of the problem becomes a lower bound for the connectivity of the network. The current result generalizes a previous result by removing a restriction on the features that are loaded into the neurons during the learning phase. © 1988 Academic Press, Inc.

1. INTRODUCTION

Learning by example has emerged as the most important question in neural networks. Clearly, a given neural network cannot just learn any function, there must be some restrictions on which networks can learn which functions. One obvious restriction, which is independent of the learning aspect, is that the network must be big enough to accommodate the circuit complexity of the function it will eventually simulate. A restriction that arises merely from the fact that the network is expected to *learn* the function, rather than being purposely designed for the function is reported in (Abu-Mostafa, 1988). The restriction imposes a lower bound on the connectivity of the network (number of synapses per neuron). In this paper, we describe a generalization of this result by removing one of the requirements on the learning mechanism. Instead of requiring that the

* Research supported by the Air Force Office of Scientific Research under Grant AFOSR-86-0296.

training sample itself be loaded directly into the neurons, we now allow arbitrary features to be extracted from the sample and loaded into the neurons. This also implies that the number of neurons can be very large with respect to the number of bits in each sample.

However, our generalized result still assumes a local-learning mechanism. The local-learning assumption allows only local information to be used by each neuron in its learning effort. The assumption cannot be completely removed since a powerful learning mechanism can be designed that will find one of the low-connectivity (e.g., two-input-NAND-gate) circuits that fits all the training samples, perhaps by exhaustive search. Local learning is a strong assumption that excludes sophisticated learning mechanisms used in neural-network models.

The lower bound on the connectivity of the network is given in terms of the *entropy* of the environment that provides the training samples. Entropy is a quantitative measure of the disorder or randomness in an environment or, equivalently, the amount of information needed to specify the environment. In Section 2, we shall introduce the formal definitions and results, but we start here with an informal exposition of the ideas involved.

The environment in our model produces patterns represented by N bits $\mathbf{x} = x_1, \dots, x_N$ (pixels in the picture of a visual scene if you will). Only h different patterns can be generated by a given environment, where $h < 2^N$ (the entropy is essentially $\log_2 h$). No knowledge is assumed about which patterns the environment can generate, only that there are h of them. In the learning process, a number of sample patterns are generated at random from the environment. A large number of binary features are extracted from each sample and input to the network, one feature per neuron. The network uses this information to set its internal parameters and gradually tune itself to this particular environment. Because of the network architecture, each neuron knows only its own bit and the bits of the neurons it is directly connected to by a synapse. Hence, the learning rules are local: a neuron does not have the benefit of the entire global pattern that is being learned.

After the learning process has taken place, each neuron is ready to perform a function *defined by what it has learned*. The collective interaction of the functions of the neurons is what defines the overall function of the network. The main result of this paper is that (roughly speaking) if the connectivity of the network is less than the entropy of the environment, the network cannot learn about the environment. The idea of the proof is to show that if the connectivity is small, the final function of each neuron is independent of the environment, and hence to conclude that the overall network has accumulated no information about the environment it is supposed to learn about.

2. LOCAL-LEARNING NETWORKS

A neural network can be described as an undirected graph (the vertices are the neurons and the edges are the synapses). Label the neurons $1, \dots, N$. Each neuron can store one bit at a time, but it also has access to those bits stored by the other neurons to which it is directly connected by a synapse. By local learning, we mean that the adjustments a neuron makes when the network is loaded with a training sample will depend only on the bits it has access to, namely, its own bit and the bits of its neighbors. In other words, the neuron does not have the benefit of the global picture in its effort to learn, just the bits it can see locally.

During the learning phase, an unknown environment provides a sequence of training samples to the network. The environment is a subset $e \subseteq \{0, 1\}^N$ (each $\mathbf{x} \in e$ is a possible sample from the environment). When the environment produces a sample \mathbf{x} , binary features f_1, \dots, f_N are extracted from \mathbf{x} and loaded into the neurons $1, \dots, N$, respectively (a feature is a function $f_i: \{0, 1\}^N \rightarrow \{0, 1\}$). For a given network, the features f_1, \dots, f_N are arbitrary but fixed, and N (the number of neurons) can be much larger than N (the number of bits in a sample), e.g., N can be superexponential in N .

As the samples from the unknown environment e come in, each neuron sees the subset of features carried by itself and its neighbors. Consider an arbitrary neuron that sees K features (we will assume $K \leq N \leq N$ throughout), and relabel $1, \dots, N$ to make these features f_1, \dots, f_K . Based on the values f_1, \dots, f_K assume as \mathbf{x} varies over e , the neuron is supposed to learn about the environment such that, after the learning phase is over, the collective behavior of the network is tuned to the environment e that provided the samples. How the neurons absorb the learning information and what computation the network is supposed to perform eventually are left deliberately unspecified. The arguments in this paper are based on the lack of information rather than the failure to use information.

The connectivity is measured by the parameter K . Since our result is asymptotic in N , we will specify K as a function of N ; $K = \alpha N$ where $\alpha = \alpha(N)$ satisfies $\lim_{N \rightarrow \infty} \alpha(N) = \alpha_0$ ($0 < \alpha_0 < 1$). To formalize the concept of unknown environment, we will consider the ensemble of environments \mathcal{E} of fixed entropy (Abu-Mostafa, 1986)

$$\mathcal{E} = \mathcal{E}(N) = \{e \subseteq \{0, 1\}^N \mid |e| = h\},$$

where $h = 2^{\beta N}$ (the entropy is essentially $\log_2 h = \beta N$) and $\beta = \beta(N)$ satisfies $\lim_{N \rightarrow \infty} \beta(N) = \beta_0$ ($0 < \beta_0 < 1$). The probability distribution on \mathcal{E} is uniform; any environment $e \in \mathcal{E}$ is as likely to occur as any other.

The neuron sees only the K (fixed but arbitrary) functions f_1, \dots, f_K of each \mathbf{x} generated by the environment e . For each e , we define the function $n: \{0, 1\}^K \rightarrow \{0, 1, 2, \dots\}$ where

$$n(a_1, \dots, a_K) = |\{\mathbf{x} \in e \mid f_k(\mathbf{x}) = a_k \text{ for } k = 1, \dots, K\}|$$

and the normalized version

$$\nu(a_1, \dots, a_K) = \frac{n(a_1, \dots, a_K)}{h}.$$

The function ν describes the relative frequency of occurrence for each of the 2^K binary vectors $f_1(\mathbf{x}), \dots, f_K(\mathbf{x})$ as \mathbf{x} runs through all h vectors in e . In other words, ν specifies the nonlinear projection of e as seen by the neuron. Clearly, $\nu(\mathbf{a}) \geq 0$ for all $\mathbf{a} \in \{0, 1\}^K$ and $\sum_{\mathbf{a} \in \{0, 1\}^K} \nu(\mathbf{a}) = 1$.

Corresponding to two environments e_1 and e_2 , we will have two functions ν_1 and ν_2 . If ν_1 is not distinguishable from ν_2 , the neuron cannot tell the difference between e_1 and e_2 . The distinguishability between ν_1 and ν_2 can be measured by

$$d(\nu_1, \nu_2) = \frac{1}{2} \sum_{\mathbf{a} \in \{0, 1\}^K} |\nu_1(\mathbf{a}) - \nu_2(\mathbf{a})|.$$

The range of $d(\nu_1, \nu_2)$ is $0 \leq d(\nu_1, \nu_2) \leq 1$, where "0" corresponds to complete indistinguishability while "1" corresponds to maximum distinguishability. The main result of this paper is to relate this distinguishability to how the connectivity of the network compares with the entropy of the environment.

3. MAIN RESULT

Let e_1 and e_2 be independently selected environments from \mathcal{E} according to the uniform probability distribution. $d(\nu_1, \nu_2)$ is now a random variable, and we are interested in the expected value $E(d(\nu_1, \nu_2))$. The case where $E(d(\nu_1, \nu_2)) = 0$ corresponds to the neuron getting no information about the environment, while the case where $E(d(\nu_1, \nu_2)) = 1$ corresponds to the neuron getting maximum information. $E(d(\nu_1, \nu_2))$ depends, among other things, on the choice of the features f_1, \dots, f_K . For example, a poor choice of the f_k 's as constant functions forces $E(d(\nu_1, \nu_2))$ to be zero regardless of K . For which values of K does there exist a choice of the f_k 's that makes $E(d(\nu_1, \nu_2))$ close to 1, and for which values is $E(d(\nu_1, \nu_2))$ close to 0 for all choices of the f_k 's? The theorem predicts these extremes depending on how the connectivity (represented by α_0 in the limit) compares with the entropy (represented by β_0 in the limit).

THEOREM.

1. If $\alpha_0 > \beta_0$, then for every N there exist functions f_1, \dots, f_K , such that $\lim_{N \rightarrow \infty} E(d(\nu_1, \nu_2)) = 1$.

2. If $\alpha_0 < \beta_0$, then for all functions f_1, \dots, f_K for all N , $\lim_{N \rightarrow \infty} E(d(\nu_1, \nu_2)) = 0$.

Proof. 1. We shall take the functions f_1, \dots, f_K to be the simple projection functions $f_k(x_1, \dots, x_k, \dots, x_N) = x_k$. Thus the neuron sees the first K bits x_1, \dots, x_K of the sample $\mathbf{x} = x_1, \dots, x_N$. We start with some basic properties about the ensemble of environments \mathcal{E} . Since the probability distribution on \mathcal{E} is uniform and since $|\mathcal{E}| = \binom{2^N}{h}$, we have

$$\Pr(e) = \binom{2^N}{h}^{-1}$$

which is equivalent to generating e by choosing h elements $\mathbf{x} \in \{0, 1\}^N$ with uniform probability (without replacement). It follows that

$$\Pr(\mathbf{x} \in e) = \frac{h}{2^N}$$

while for $\mathbf{x}_1 \neq \mathbf{x}_2$,

$$\Pr(\mathbf{x}_1 \in e, \mathbf{x}_2 \in e) = \frac{h}{2^N} \times \frac{h-1}{2^N-1}$$

and so on.

The functions n and ν are defined on K -bit vectors. For the above choice of the functions f_1, \dots, f_K , the statistics of $n(\mathbf{a})$ (a random variable for fixed \mathbf{a}) are independent of \mathbf{a} ,

$$\Pr(n(\mathbf{a}_1) = m) = \Pr(n(\mathbf{a}_2) = m),$$

which follows from the symmetry with respect to each bit of \mathbf{a} . The same holds for the statistics of $\nu(\mathbf{a})$. The expected value $E(n(\mathbf{a})) = h2^{-K}$ (h objects going into 2^K cells), hence $E(\nu(\mathbf{a})) = 2^{-K}$.

We expand $E(d(\nu_1, \nu_2))$ as

$$\begin{aligned} E(d(\nu_1, \nu_2)) &= E\left(\frac{1}{2} \sum_{\mathbf{a} \in \{0,1\}^K} |\nu_1(\mathbf{a}) - \nu_2(\mathbf{a})|\right) \\ &= \frac{1}{2h} \sum_{\mathbf{a} \in \{0,1\}^K} E(|n_1(\mathbf{a}) - n_2(\mathbf{a})|) \\ &= \frac{2^K}{2h} E(|n_1 - n_2|), \end{aligned}$$

where n_1 and n_2 denote $n_1(0, \dots, 0)$ and $n_2(0, \dots, 0)$, respectively, and the last step follows from the fact that the statistics of $n_1(\mathbf{a})$ and $n_2(\mathbf{a})$ are independent of \mathbf{a} . Therefore, to prove the first part of the theorem, we assume $\alpha_0 > \beta_0$ and evaluate $E(|n_1 - n_2|)$ for large N . Let n denote $n(0, \dots, 0)$, and consider $\Pr(n = 0)$. For n to be zero, all 2^{N-K} strings \mathbf{x} of N bits starting with K 0's must *not* be in the environment e . Hence

$$\Pr(n = 0) = \left(1 - \frac{h}{2^N}\right) \left(1 - \frac{h}{2^N - 1}\right) \cdots \left(1 - \frac{h}{2^N - 2^{N-K} + 1}\right),$$

where the first term is the probability that $0, \dots, 00 \notin e$, the second term is the probability that $0, \dots, 01 \notin e$ given that $0, \dots, 00 \notin e$, and so on:

$$\begin{aligned} &\geq \left(1 - \frac{h}{2^N - 2^{N-K}}\right)^{2^{N-K}} \\ &= (1 - h2^{-N}(1 - 2^{-K})^{-1})^{2^{N-K}} \\ &\geq (1 - 2h2^{-N})^{2^{N-K}} \\ &\geq 1 - 2h2^{-N}2^{N-K} \\ &= 1 - 2h2^{-K}. \end{aligned}$$

Hence, $\Pr(n_1 = 0) = \Pr(n_2 = 0) = \Pr(n = 0) \geq 1 - 2h2^{-K}$. However, $E(n_1) = E(n_2) = h2^{-K}$. Therefore,

$$\begin{aligned} E(|n_1 - n_2|) &= \sum_{i=0}^h \sum_{j=0}^h \Pr(n_1 = i, n_2 = j) |i - j| \\ &= \sum_{i=0}^h \sum_{j=0}^h \Pr(n_1 = i) \Pr(n_2 = j) |i - j| \\ &\geq \sum_{j=0}^h \Pr(n_1 = 0) \Pr(n_2 = j) j \\ &\quad + \sum_{i=0}^h \Pr(n_1 = i) \Pr(n_2 = 0) i \end{aligned}$$

which follows by throwing away all the terms where neither i nor j is zero (the term where both i and j are zero appears twice for convenience, but this term is zero anyway):

$$\begin{aligned} &= \Pr(n_1 = 0)E(n_2) + \Pr(n_2 = 0)E(n_1) \\ &\geq 2(1 - 2h2^{-K})h2^{-K}. \end{aligned}$$

Substituting this estimate in the expression for $E(d(\nu_1, \nu_2))$, we get

$$\begin{aligned} E(d(\nu_1, \nu_2)) &= \frac{2^K}{2h} E(|n_1 - n_2|) \\ &\geq \frac{2^K}{2h} \times 2(1 - 2h2^{-K})h2^{-K} \\ &= 1 - 2h2^{-K} \\ &= 1 - 2 \times 2^{(\beta-\alpha)N}. \end{aligned}$$

Since $\alpha_0 > \beta_0$ by assumption, this lower bound goes to 1 as N goes to infinity. Since 1 is also an upper bound for $d(\nu_1, \nu_2)$ (and hence an upper bound for the expected value $E(d(\nu_1, \nu_2))$), $\lim_{N \rightarrow \infty} E(d(\nu_1, \nu_2))$ must be 1.

2. Assume $\alpha_0 < \beta_0$, and consider arbitrary functions f_1, \dots, f_K . Define

$$\bar{n}(\mathbf{a}) = \frac{h}{2^N} |\{\mathbf{x} \in \{0, 1\}^N \mid f_k(\mathbf{x}) = a_k \text{ for } k = 1, \dots, K\}|.$$

We expand $E(d(\nu_1, \nu_2))$

$$\begin{aligned} E(d(\nu_1, \nu_2)) &= \frac{1}{2h} \sum_{\mathbf{a} \in \{0,1\}^K} E(|n_1(\mathbf{a}) - n_2(\mathbf{a})|) \\ &= \frac{1}{2h} \sum_{\mathbf{a} \in \{0,1\}^K} E(|n_1(\mathbf{a}) - \bar{n}(\mathbf{a}) - (n_2(\mathbf{a}) - \bar{n}(\mathbf{a}))|) \\ &\leq \frac{1}{2h} \sum_{\mathbf{a} \in \{0,1\}^K} E(|n_1(\mathbf{a}) - \bar{n}(\mathbf{a})| + |n_2(\mathbf{a}) - \bar{n}(\mathbf{a})|) \\ &= \frac{1}{2h} \sum_{\mathbf{a} \in \{0,1\}^K} E(|n_1(\mathbf{a}) - \bar{n}(\mathbf{a})|) + E(|n_2(\mathbf{a}) - \bar{n}(\mathbf{a})|) \\ &= \frac{1}{h} \sum_{\mathbf{a} \in \{0,1\}^K} E(|n(\mathbf{a}) - \bar{n}(\mathbf{a})|). \end{aligned}$$

The statistics of $n(\mathbf{a})$ now depend on \mathbf{a} since the functions f_1, \dots, f_K are arbitrary. To evaluate $E(|n(\mathbf{a}) - \bar{n}(\mathbf{a})|)$, we first show that $\bar{n}(\mathbf{a}) = E(n(\mathbf{a}))$, then estimate the variance of $n(\mathbf{a})$ and use the fact that $E(|n(\mathbf{a}) - E(n(\mathbf{a}))|) \leq \sqrt{\text{var}(n(\mathbf{a}))}$. We write

$$n(\mathbf{a}) = \sum_{\mathbf{x} \in \{0,1\}^N} \delta(\mathbf{x}, \mathbf{a}) \delta(\mathbf{x}),$$

where $\delta(\mathbf{x}, \mathbf{a}) = 1$ if $f_k(\mathbf{x}) = a_k$ for $k = 1, \dots, K$ and is zero otherwise, and $\delta(\mathbf{x}) = 1$ if $\mathbf{x} \in e$ and is zero otherwise (while $\delta(\mathbf{x}, \mathbf{a})$ is fixed for given \mathbf{x} and \mathbf{a} , $\delta(\mathbf{x})$ is a random variable for a given \mathbf{x}). Hence

$$E(n(\mathbf{a})) = \sum_{\mathbf{x} \in \{0,1\}^N} \delta(\mathbf{x}, \mathbf{a}) E(\delta(\mathbf{x})).$$

The expected value of $\delta(\mathbf{x})$ is $\Pr(\mathbf{x} \in e) = h/2^N$. Factoring this out, we are left with $\sum_{\mathbf{x} \in \{0,1\}^N} \delta(\mathbf{x}, \mathbf{a})$ which equals $|\{\mathbf{x} \in \{0, 1\}^N \mid f_k(\mathbf{x}) = a_k \text{ for } k = 1, \dots, K\}|$, hence $E(n(\mathbf{a}))$ indeed equals $\bar{n}(\mathbf{a})$.

Since $\text{var}(n(\mathbf{a})) = E((n(\mathbf{a}))^2) - (E(n(\mathbf{a})))^2$, we need an estimate for $E((n(\mathbf{a}))^2)$:

$$\begin{aligned} E((n(\mathbf{a}))^2) &= E\left(\sum_{\mathbf{x}_1 \in \{0,1\}^N} \sum_{\mathbf{x}_2 \in \{0,1\}^N} \delta(\mathbf{x}_1, \mathbf{a}) \delta(\mathbf{x}_2, \mathbf{a}) \delta(\mathbf{x}_1) \delta(\mathbf{x}_2)\right) \\ &= \sum_{\mathbf{x}_1} \sum_{\mathbf{x}_2} \delta(\mathbf{x}_1, \mathbf{a}) \delta(\mathbf{x}_2, \mathbf{a}) E(\delta(\mathbf{x}_1) \delta(\mathbf{x}_2)). \end{aligned}$$

For the "diagonal" terms ($\mathbf{x}_1 = \mathbf{x}_2$), we get $\sum_{\mathbf{x}} \delta(\mathbf{x}, \mathbf{a}) E(\delta(\mathbf{x}))$ (since $\delta^2 = \delta$), which equals $\bar{n}(\mathbf{a})$. For the "off-diagonal" terms ($\mathbf{x}_1 \neq \mathbf{x}_2$), we get

$$\begin{aligned} &\sum_{\mathbf{x}_1} \sum_{\mathbf{x}_2 \neq \mathbf{x}_1} \delta(\mathbf{x}_1, \mathbf{a}) \delta(\mathbf{x}_2, \mathbf{a}) E(\delta(\mathbf{x}_1) \delta(\mathbf{x}_2)) \\ &= \sum_{\mathbf{x}_1} \sum_{\mathbf{x}_2 \neq \mathbf{x}_1} \delta(\mathbf{x}_1, \mathbf{a}) \delta(\mathbf{x}_2, \mathbf{a}) \Pr(\mathbf{x}_1 \in e, \mathbf{x}_2 \in e) \\ &= \sum_{\mathbf{x}_1} \sum_{\mathbf{x}_2 \neq \mathbf{x}_1} \delta(\mathbf{x}_1, \mathbf{a}) \delta(\mathbf{x}_2, \mathbf{a}) \frac{h(h-1)}{2^N(2^N-1)} \\ &= \sum_{\mathbf{x}_1} \sum_{\mathbf{x}_2} \delta(\mathbf{x}_1, \mathbf{a}) \delta(\mathbf{x}_2, \mathbf{a}) \frac{h(h-1)}{2^N(2^N-1)} - \sum_{\mathbf{x}} \delta(\mathbf{x}, \mathbf{a}) \frac{h(h-1)}{2^N(2^N-1)}. \end{aligned}$$

The last step follows by adding and subtracting the missing terms of the double summation. Noting that $\bar{n}(\mathbf{a}) = (h/2^N) \sum_{\mathbf{x}} \delta(\mathbf{x}, \mathbf{a})$, this can be rewritten as

$$\frac{2^N(h-1)}{h(2^N-1)} (\bar{n}(\mathbf{a}))^2 - \frac{h-1}{2^N-1} \bar{n}(\mathbf{a}).$$

Putting the contributions from the diagonal and off-diagonal terms together, we get

$$\begin{aligned}
E((n(\mathbf{a}))^2) &= \bar{n}(\mathbf{a}) + \frac{2^N(h-1)}{h(2^N-1)} (\bar{n}(\mathbf{a}))^2 - \frac{h-1}{2^N-1} \bar{n}(\mathbf{a}) \\
&= \frac{2^N(h-1)}{h(2^N-1)} (\bar{n}(\mathbf{a}))^2 + \frac{2^N-h}{2^N-1} \bar{n}(\mathbf{a}) \\
\text{var}(n(\mathbf{a})) &= E((n(\mathbf{a}))^2) - (E(n(\mathbf{a})))^2 \\
&= \frac{2^N(h-1)}{h(2^N-1)} (\bar{n}(\mathbf{a}))^2 + \frac{2^N-h}{2^N-1} \bar{n}(\mathbf{a}) - (\bar{n}(\mathbf{a}))^2 \\
&= \frac{h-2^N}{h(2^N-1)} (\bar{n}(\mathbf{a}))^2 + \frac{2^N-h}{2^N-1} \bar{n}(\mathbf{a}) \\
&= \frac{2^N-h}{2^N-1} \bar{n}(\mathbf{a}) \left(1 - \frac{1}{h} \bar{n}(\mathbf{a})\right) \\
&\leq \frac{2^N-h}{2^N-1} \bar{n}(\mathbf{a}) \\
&\leq \bar{n}(\mathbf{a}).
\end{aligned}$$

Thus we have $E(|n(\mathbf{a}) - \bar{n}(\mathbf{a})|) \leq \sqrt{\text{var}(n(\mathbf{a}))} \leq \sqrt{\bar{n}(\mathbf{a})}$. Now, we rewrite the estimate for $E(d(\nu_1, \nu_2))$:

$$\begin{aligned}
E(d(\nu_1, \nu_2)) &\leq \frac{1}{h} \sum_{\mathbf{a} \in \{0,1\}^K} E(|n(\mathbf{a}) - \bar{n}(\mathbf{a})|) \\
&\leq \frac{1}{h} \sum_{\mathbf{a} \in \{0,1\}^K} \sqrt{\bar{n}(\mathbf{a})}.
\end{aligned}$$

The values of the individual $\bar{n}(\mathbf{a})$ will depend on the choice of f_1, \dots, f_K . However, $\sum_{\mathbf{a} \in \{0,1\}^K} \bar{n}(\mathbf{a})$ always equals h (from the definition of $\bar{n}(\mathbf{a})$). Therefore, one can obtain an upper bound for $E(d(\nu_1, \nu_2))$ by maximizing $\sum_{\mathbf{a} \in \{0,1\}^K} \sqrt{\bar{n}(\mathbf{a})}$ subject to $\sum_{\mathbf{a} \in \{0,1\}^K} \bar{n}(\mathbf{a}) = h$. The maximum occurs when all $\bar{n}(\mathbf{a})$ are equal ($= h2^{-K}$). Hence, $E(d(\nu_1, \nu_2)) \leq (1/h)2^K \sqrt{h2^{-K}} = \sqrt{2^K/h} = 2^{(1/2)(\alpha-\beta)N}$. Since $\alpha_0 < \beta_0$ by assumption, this upper bound goes to 0 as N goes to infinity. Since 0 is also a lower bound for $d(\nu_1, \nu_2)$ (and hence a lower bound for the expected value $E(d(\nu_1, \nu_2))$), $\lim_{N \rightarrow \infty} E(d(\nu_1, \nu_2))$ must be 0. ■

4. CONCLUSION

We have shown that, under the assumption of local learning, each neuron must have at least a certain number of synapses in order to be able

to distinguish between environments based on the statistics of information it sees. While the result is expressed as a limit, it is seen in the proof that the rate of convergence to this limit is exponential in N , the dimensionality of the problem. Further work should address the weakening of the local-learning assumption, perhaps by restricting the amount of global information flow or by restricting the ability of the neuron to make use of the information it sees (e.g., by modeling its learning mechanism as a finite-state machine).

ACKNOWLEDGMENT

I thank Dr. Edward C. Posner for his assistance.

REFERENCES

- ABU-MOSTAFA, Y. S. (1986), The complexity of information extraction, *IEEE Trans. Inform. Theory* **IT-32**, 513-525.
- ABU-MOSTAFA, Y. S. (1988), Connectivity versus Entropy, in "Neural Information Processing Systems" (D. Anderson, Ed.), American Institute of Physics.