CONF-9606113--2

Function Estimation by Feedforward Sigmoidal Networks with Bounded Weights †

Nageswara S.V. Rao
Vladimir Protopopescu
Center for Engineering Systems Advanced Research
Oak Ridge National Laboratory
Oak Ridge, Tennessee 37831-6364

Hongzhu Qiao
Department of Mathematics and Physics
Fort Valley State College
Fort Valley, GA 31030

Submitted to: *Ninth Conference on Computational Learning Theory*, Desenzano del Garda, Italy, June 28 - July 1, 1996.

†Research sponsored by the Engineering Research Program of the Office of Basic Energy Sciences, of the U.S. Department of Energy, under Contract No. DE-AC05-96OR22464 with Lockheed Martin Energy Research Corp.

Function Estimation by Feedforward Sigmoidal Networks with Bounded Weights [1]

Nageswara S. V. Rao † and V. Protopopescu
Center for Engineering Systems Advanced Research
Oak Ridge National Laboratory
Oak Ridge, Tennessee 37831-6364
{raons,protopopesva} @ornl.gov

Hongzhu Qiao
Department of Mathematics and Physics
Fort Valley State College
Fort Valley, GA 31030
hqiao@gauss.math.usf.edu

Summary

We address the problem of PAC learning functions $f : [0,1]^d \mapsto [-K, K]$ based on an iid sample generated according to an unknown distribution, by using feedforward sigmoidal networks. We use two basic properties of the neural networks with bounded weights, namely: (a) they form a Euclidean class, and (b) for hidden units of the form $\tanh(\gamma z)$ they are Lipschitz functions. Either property yields sample sizes for PAC function learning under any Lipschitz cost function. The sample size based on the first property is tighter compared to the known bounds based on VC-dimension. The second estimate yields a sample size that can be conveniently adjusted by a single parameter, γ, related to the hidden nodes.

1 Introduction

The problem of learning functions in the Probably and Approximately Correct (PAC) learning framework of Valiant [23] continues to generate significant interest and activity [1, 5, 6, 10, 3]. Initial efforts were focussed on indicator functions and functions on simpler domains [14] with increasing attention being paid to general real functions [4, 21]. Recent results establish that a function that achieves small empirical error on an independently and identically distributed (iid) sample yields a PAC approximation, under the finiteness of a combinatorial parameter such as the fat-shattering index [5, 2].

In this paper, we employ feedforward neural networks to solve the function estimation problem based on random samples. Artificial neural networks have been extensively applied to a variety of applications involving function estimation [19, 8, 20] based on VC-dimension and related parameters. The performance of neural networks based on other parameters is less clear, and a better understanding of their learning capabilities is needed for improving their efficiency in applications. The PAC paradigm of Valiant [23] and the empirical risk minimization method of Vapnik [25] enable us to characterize and quantify the performance of neural networks as function estimators in terms of more general and/or alternative parameters.

We are given iid points $X_1, X_2, \ldots, X_n$ from $[0,1]^d$ according to an unknown distribution P_X and the corresponding values of an unknown function $f : [0,1]^d \mapsto [-K, K]$ chosen from a family $\mathcal{F}$. Throughout the paper, X and x denote the random and deterministic variables respectively, and it is assumed that all functions satisfy the required measurability conditions. We consider the problem of estimating an approximation to f from the class of feedforward neural networks with a single hidden layer of l sigmoid units. We assume that each connection weight is chosen from $[-A, A]$, for finite $A > 0$. Such assumption is generally made in nonparametric estimation using feedforward neural networks [26, 27] and is reasonable since in practical applications the weights are bounded. These neural networks constitute a family of functions $\mathcal{F}_{\mathcal{A}} = \{f_w : w \in [-A, A]^{l(d+2)}\}$, where f_w corresponds to a neural network with a parameter vector w.

Consider a bounded *cost function* $\Theta : \mathcal{F}_{\mathcal{A}} \mapsto \mathcal{G}$ where $\mathcal{G} = \{g : [0,1]^d \mapsto \Re\}$, i. e. for any $f_w \in \mathcal{F}_{\mathcal{A}}$, the function $\Theta(f_w) : [0,1]^d \mapsto \Re$ specifies the cost of approximating the unknown $f(x)$ by the estimated $f_w(x)$ for any $x \in [0,1]^d$. For convenience, $\Theta(f_w)(x)$ for $x \in [0,1]^d$ will be denoted by $\Theta(x, f_w(x))$ with an abuse of notation. Note that the well-known square error formulation corresponds to the special case

[1] Research sponsored by the Engineering Research Program of the Office of Basic Energy Sciences, of the U.S. Department of Energy, under Contract No. DE-AC05-96OR22464 with Lockheed Martin Energy Research Corp.

†Address all correspondence to the first author; email: raons@onrl.gov

$\Theta(x, f_w(x)) = (f(x) - f_w(x))^2$. Consider the *expected cost* of approximating f by f_w given by

$$I(f_w) = \int_{[0,1]^d} \Theta(X, f_w(X)) dP_X.$$

Let $f_w^* \in \mathcal{F}_A$ minimize $I(.)$, where f_w^* is called the *best expected neural network* in $\mathcal{F}_A$. Consider the *empirical cost* given by

$$\hat{I}(f_w) = \frac{1}{n} \sum_{i=1}^{n} \Theta(X_i, f_w(X_i)) .$$

which is minimized, say, at $\hat{f}_w \in \mathcal{F}_A$; $\hat{f}_w$ is called the *best empirical neural network* in $\mathcal{F}_A$. We investigate the conditions under which, given a sufficiently large sample of size n, we can guarantee that

$$P_X^n[I(\hat{f}_w) - I(f_w^*) > \epsilon] < \delta$$

for given $\epsilon > 0$ and $0 < \delta < 1$, where P_X^n is the product measure on the set of all n-samples. To this end, we identify and utilize two basic properties, namely:

(a) **Vector Space Property:** $\mathcal{F}_A$ can be embedded in a vector space, which implies that it forms an Euclidean class [15]. For these results we only need the boundedness of weights in the output layer as is commonly assumed for consistency results (Lugosi and Zeger [12]). To simplify the presentation though, we shall assume that all weights are bounded.

(b) **Smoothness Property:** One of the most applied feedforward networks consists of a single hidden layer of l nodes, each of the form $\sigma(z) = \tanh(\gamma z)$, $0 < \gamma < \infty$, $z \in \Re$. For these networks we exploit the (Lipschitz) smoothness properties to obtain sample size estimates. In this case, the sample size can be adjusted by varying γ alone, while all other parameters are fixed.

We estimate the required sample size directly in terms of the parameters of the neural network, namely: (i) number of parameters, $l(d+2)$, (ii) the bound on weights, A, (iii) the slope of the sigmoid, γ, and (iv) the Lipschitz constant, L_Θ, of $\Theta(.)$.

The following aspects distinguish our work from the existing ones:

(a) The general approach of relating algebraic or smoothness properties to the sample sizes for the estimators was proposed in Dudley [7]. Here we illustrate the applicability of these ideas to neural networks and determine the underlying constants in the sample sizes which were not completely specified in [7].

(b) Sharper bounds are obtained on the sample size by using *Euclidean class* formulation as opposed to VC-dimension based estimate [8, 19]. A similar result was established for the special case $\mathcal{F} = \mathcal{F}_A$ by a very different approach by Shawe-Taylor [20];

(c) Easier and simpler proofs are provided for finiteness of VC-dimension and Euclidean parameters by making use of existing results from the empirical processes (Nolan and Pollard [15]). Our sample sizes are closely related to those by Lugosi and Zeger [12], but our derivation is easier and more direct. Moreover, it establishes a connection between the underlying computational problem and the *linear problems* studied in the area of information-based complexity (Traub *et al.* [22]).

(d) The smoothness (Lipschitz) properties of neural networks are utilized to obtain an alternate sample size estimate. We are unaware of such results in the mainstream PAC literature (e. g. [20]); smoothness properties, however, have been extensively used in non-parametric estimation to establish asymptotic results (Prakasa Rao [17]), and more recently in obtaining finite sample results (Rao and Protopopescu [18]); and

(e) Finite sample results are obtained for a class of Lipschitz cost functions (see Section 4 for a precise definition), which includes the mean square error.

Preliminaries are presented in Section 2. The basic covering properties of neural networks are presented in Section 3; two characterizations based on Euclidean class and Lipschitz functions are presented in Sections 3.1 and 3.2, respectively. Sample size estimates for function estimation problem are presented in Section 4.

2 Preliminaries

We consider a feedforward network with a single hidden layer of l nodes and a single output node. The output of the jth hidden node is $\sigma(b_j^T x + t_j)$, where $x \in [0,1]^d$, $b_j \in \Re^d$, $t_j \in \Re$, and the nondecreasing $\sigma : \Re \mapsto [-1,+1]$ is called the *activation function*. The output of the network corresponding to input x is given by

$$f_w(x) = \sum_{j=1}^{l} a_j \sigma(b_j^T x + t_j)$$

where $w = (w_1, w_2, \ldots, w_{l(d+2)})$ is the *weight vector* of the network consisting of $a_1, a_2, \ldots, a_l, b_{11}, b_{12}, \ldots, b_{1d}$, $b_{l1}, \ldots b_{ld}$, and $t_1, t_2, \ldots, t_l$. Let the set of sigmoidal networks with bounded weights be denoted by

$$\mathcal{F}_A = \{f_w : w \in [-A, A]^{l(d+2)}, 0 < A < \infty\}. \tag{2.1}$$

We consider a subclass of $\mathcal{F}_A$ where each hidden unit is of the particular form $\sigma(z) = \tanh(\gamma z)$, for $0 < \gamma < \infty$, namely

$$\mathcal{F}_A^\gamma = \{f_w : w \in [-A, A]^{l(d+2)}, \sigma(z) = \tanh(\gamma z)\}. \tag{2.2}$$

Let S be a set equipped with a pseudometric d. The *covering number* $N(\epsilon, d, S)$ is defined as the smallest number of closed balls of radius ϵ, and centers in S, whose union covers S.

The function class $\mathcal{F}$ has an *envelope* F if $f(x) \leq F(x)$ for all $f \in \mathcal{F}$. Let μ be a probability measure on $[0,1]^d$, X the corresponding random variable, and let $\mu(f^p) = \left(\int\limits_{x \in [0,1]^d} |f(X)|^p d\mu \right)^{1/p}$ for finite integer $p \geq 1$ and measurable function f. Now consider a probability measure μ such that $\mu(F^p) < \infty$ for any finite integer $p \geq 1$. We define the *covering number* $N_p(\epsilon, \mu, \mathcal{F}, F)$ to be the smallest cardinality for a subclass $\mathcal{F}^*$ of $\mathcal{F}$ such that

$$\min_{f^* \in \mathcal{F}^*} \mu(|f - f^*|^p) \leq \epsilon^p \mu(F^p)$$

for each $f \in \mathcal{F}$.

The class of functions $\mathcal{F}$ is *Euclidean for the envelope* F if there exist *Euclidean constants* B and V such that

$$N_1(\epsilon, \mu, \mathcal{F}, F) \leq B\epsilon^{-V}$$

for $0 < \epsilon \leq 1$, whenever $0 < \mu(F^1) < \infty$ [15].

Let $N_\infty(\epsilon, \mathcal{F}) = N(\epsilon, \| \cdot \|_\infty, \mathcal{F})$, where $\| f(x) \|_\infty = \sup\limits_{x \in [0,1]^d} |f(x)|$. Due to the boundedness of $\mathcal{F}$ we have

$$N_1(\epsilon/\mu(F), \mu, \mathcal{F}, F) \leq N_\infty(\epsilon, \mathcal{F})$$

since $\mu(|f - f^*|^p) \leq \left(\int |f(X) - f^*(X)|^p d\mu\right)^{1/p} \leq \| f - f^* \|_\infty$.

3 Covering Properties of Neural Networks

We first show that the set of feedforward neural networks constitute a Euclidean class and estimate bounds for $N_1(\epsilon, \mu, \mathcal{F}_A, F_A)$, where $F_A(x) = lA$ for $x \in [0,1]^d$. Then we exploit the Lipschitz property of the neural networks to estimate a bound on $N_\infty(\epsilon, \mathcal{F}_A^\gamma)$.

3.1 Euclidean Classes

Finiteness of Euclidean parameters of $\mathcal{F}_A$ (and hence the finiteness of VC-dimension of positivity sets of $\mathcal{F}_A$) can be directly concluded from the existing results. For example, the function class $\{a\sigma(b^T x + t) : a \in \Re, t \in \Re, b \in \Re^d\}$ can be shown to be Euclidean for constant envelope (Lemma 22, Nolan and Pollard [15]). It follows that the class $\{\sum\limits_{i=1}^{m} a_i \sigma(b_i^T x + t_i) : a_i \in [-A, A], t_i \in \Re, b_i \in \Re^d\}$ is Euclidean for constant envelope (Lemma 16, Nolan and Pollard [15]). If more general classes of neural networks are considered,

the finiteness of VC-dimension (and hence that of Euclidean parameters [9]) seems difficult to establish (MacIntyre and Sontag [13]). In order to estimate reasonable bounds for the sample sizes, however, a more detailed application of the result of Nolan and Pollard [15] is needed, in addition to the estimation of relevant parameters for the specific class $\mathcal{F}_A$.

For a family $\{S_\tau\}_{\tau\in\Gamma}$, $S_\tau \subseteq S$, and for a finite set $\{s_1, s_2, \ldots, s_n\} \subseteq S$, we have [24]:

$$\Pi_{\{S_\tau\}}(\{s_1, s_2, \ldots, s_n\}) = \{\{s_1, s_2, \ldots, s_n\} \cap S_\tau\}_{\tau\in\Gamma},$$

$$\Pi_{\{S_\tau\}}(n) = \max_{s_1, s_2, \ldots, s_n} |\Pi_{\{S_\tau\}}(\{s_1, s_2, \ldots, s_n\})|.$$

The following identity is established in [24]: $\Pi_{\{S_\tau\}}(n) = \begin{cases} 2^n & \text{if } n \le k \\ < 1.5\frac{n^k}{k!} & \text{if } n > k. \end{cases}$

The quantity k is called the *Vapnik-Chervonenkis* (VC) dimension of the family $\{S_\tau\}$, denoted by $VC(\{S_\tau\})$.

The *graph* of a function $f : E \mapsto [-K, +K]$ is defined by Pollard [16] as the subset of $E \times [-K, K]$ given by

$$G(f) = \{(x, t) : x \in E, 0 \le t \le f(x) \text{ or } f(x) \le t \le 0\}.$$

We now show that $\mathcal{F}_A$ forms a Euclidean class by embedding it in a vector space of functions defined on $[0, 1]^{d+1}$.

Lemma 3.1 *Consider the class* $\mathcal{F}_W = \{\sum_{i=1}^{l} a_i\sigma(b_i^T y) : a_i, t_i \in \Re, b_i \in \Re^{d+1}\}$. *Then* $VC(\{G(f)|f \in \mathcal{F}_W\}) \le l(d+2)$.

Proof: We show that the class $\mathcal{F}_W$ can be organized as a vector space of dimension $l(d+2)$, and then the result follows from the lemma of Dudley [7] (see also Pollard [16] and Haussler [8]). Indeed, let us define vector addition $\oplus$ and scalar multiplication $\otimes$ as follows:

$$\left(\sum_{i=1}^{l} a_i^1\sigma((b_i^1)^T y)\right) \oplus \left(\sum_{i=1}^{l} a_i^2\sigma((b_i^2)^T y)\right) = \sum_{i=1}^{l}(a_i^1 + a_i^2)\sigma((b_i^1 + b_i^2)^T y)$$

and

$$c \otimes \left(\sum_{i=1}^{l} a_i\sigma(b_i^T y)\right) = \sum_{i=1}^{l} ca_i\sigma(cb_i^T y).$$

The additive identity of the vector space is given by $a_i = 0, t_i = 0, b_{ij} = 0$ for $i = 1, 2, \ldots, l$ and $j = 1, 2, \ldots, d$. By noting that $\oplus$ operates on each of the l terms of the form $a\sigma(b^T y)$ independently, we can construct a $l(d+2)$ functions that span $\{a\sigma(b^T y) : a, t \in \Re, b \in \Re^{d+1}\}$ under the vector space operations $\oplus$ and $\otimes$. Consider the functions $\{\sigma(1_1 y), \ldots, \sigma(1_{d+1} y), \sigma(0)\}$, where $1_i y = y_i$, for $i = 1, \ldots, d+1$. Then we have

$$\begin{aligned}[c_0 \otimes \sigma(0)] \oplus [c_1 \otimes \sigma(1_1 y)] \oplus \ldots \oplus [c_{d+1} \otimes \sigma(1_{d+1} y)] \\ = (c_0 + c_1 + \ldots + c_d + c_{d+1})\sigma(c^T y)\end{aligned}$$

where $c^T = (c_1, \ldots, c_{d+1})$. Then any function $a\sigma(b^T y)$ can be generated from the above basis by choosing $c_i = b_i$, $i = 1, \ldots d+1$ and $c_0 = a - (b_1 + \ldots + b_{d+1})$. □

The number of subsets, m, of k points $(x, y) \in [0, 1]^d \times [-k, k]$ contained in the graphs of $\mathcal{F}_W$ is bounded as follows:

$$m = \begin{cases} 2^k & \text{if } k \le l(d+2) \\ < 1.5\frac{k^{l(d+2)}}{(l(d+2))!} & \text{if } k > l(d+2) \end{cases}$$

and thus m is upperbounded by $\le C_W k^{l(d+2)}$, for some C_W . We can establish the following result.

Lemma 3.2 *For the class of feedforward neural networks* $\mathcal{F}_A$ *of Eq (2.1), we have*

$$N_1(\epsilon, \mu, \mathcal{F}_A, F_A) \le D\epsilon^{-2l(d+2)},$$

where $D = \max(C_W^2, k_0)(lA)^{2l(d+2)}$ *and* k_0 *is determined by the inequality* $(1 + 4\log k)^{l(d+2)} \le k^{1/2}$ *for all* $k \ge k_0$.

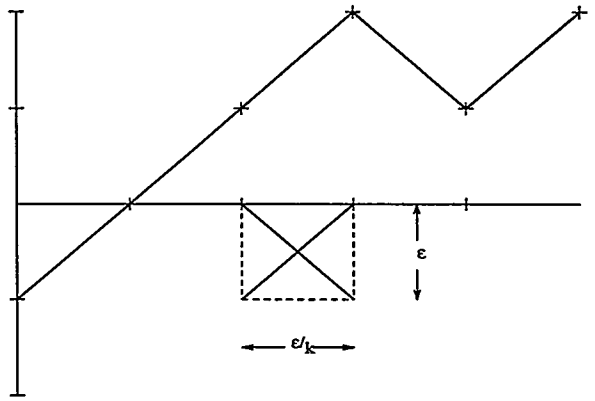

Figure 1: Illustration of cover for $d = 1$.

Proof: From the proof of Lemma 28 of Pollard [16], we have

$$VC[G(\mathcal{F}_\mathcal{A})] \leq VC(\{x \in [0,1]^d : f(x) > 0\}_{f \in \mathcal{F}_\mathcal{A}}) + 1,$$

by utilizing the symmetry property of $\mathcal{F}_\mathcal{A}$. From Lemma 3.1 and the proof of the Approximation Lemma of Pollard ([16], Lemma 25, p. 27) we have

$$N_1(\epsilon \mu F_A, \mu, \mathcal{F}_\mathcal{A}, F_A) \leq B\epsilon^{-2l(d+2)+2}$$

where $B = \max(C_W^2, k_0)$. Since $\sigma(.)$ is upperbounded by 1, we have $f_w(x) \leq \sum_{i=1}^{l} |a_i| \leq lA$. The result follows by noting that $F_A(x) \leq lA$, $\mu F_A \leq lA$ and $N_1(\epsilon lA, \mu, \mathcal{F}_\mathcal{A}, F_A) \leq N_1(\epsilon \mu F_A, \mu, \mathcal{F}_\mathcal{A}, F_A)$.□

The above lemma is valid under the slightly weaker condition that only a_i's are chosen from $[-A, A]$, since $\sigma(.) \leq 1$ and the boundedness of a_i's is sufficient to show that the envelope exists.

Lugosi and Zeger [12] provide an alternative bound namely $N_1(\epsilon, \mu, \mathcal{F}_\mathcal{A}, F_A) \leq \left(\frac{4e(l+1)lA}{\epsilon}\right)^{l(2d+3)+1}$ using a series of lemmas. Our result is more direct and also explicitly specifies the vector space structure of the neural networks, which could be of independent interest.

3.2 Lipschitz functions

We first estimate a bound for the cover size for a general class of Lipschitz functions.

Lemma 3.3 *Let $\mathcal{F}_k = \{f_k : [0,1]^d \mapsto \Re\}$ denote a set of Lipschitz functions with Lipschitz constant k, i. e. for every $f_w \in \mathcal{F}_k$, we have $|f_w(x) - f_w(y)| \leq k \parallel x - y \parallel$, where $\parallel x - y \parallel = \max_{i=1}^{d} |x_i - y_i|$. Then*

$$N_\infty(\epsilon, \mathcal{F}_k) \leq \frac{2k}{\epsilon} 2^{\left\{\frac{k}{\epsilon}\left[\left(\frac{k}{\epsilon}-1\right)^{d-1}+1\right]\right\}}.$$

Proof: Let $N_\infty^i(\epsilon, \mathcal{F})$ be the size of the cover for functions defined on the domain $[0,1]^i$ such that $N_\infty^d(\epsilon, \mathcal{F}) = N_\infty(\epsilon, \mathcal{F})$. We first consider the case $d = 1$, for which we show that

$$N_\infty^1(\epsilon, \mathcal{F}_k) \leq \frac{2k}{\epsilon} 2^{k/\epsilon}.$$

Decompose the domain $[0,1] \times [-k, k]$ into $2k^2/\epsilon^2$ cells of size $\epsilon/k \times \epsilon$ (see Fig. 1). We generate a function by following a sequence of diagonals of cells that share endpoints from left to right as illustrated in Fig. 1. Each such sequence defines a function in $\mathcal{F}_k$ defined on $[0,1]$, and any function $f \in \mathcal{F}_k$ will be within ϵ of one of the functions constructed above in the $\parallel . \parallel_\infty$ norm. The number of functions of such kind yields an upper bound for $N_\infty^1(\epsilon, \mathcal{F}_k)$ as follows. Let $M_i^1(\epsilon, \mathcal{F}_k)$ denote the functions defined on the domain $[0, i\epsilon/k]$ such that $M_{k/\epsilon}^1(\epsilon, \mathcal{F}_k) = N_\infty^1(\epsilon, \mathcal{F}_k)$. Now note that:

(a) $M_1^1(\epsilon, \mathcal{F}_k) \leq 4k/\epsilon$, since there are at most two functions for each cell; and

(b) $M_{i+1}^1(\epsilon, \mathcal{F}_k) \leq 2M_i(\epsilon, \mathcal{F}_k)$, since any function on $[0, i\epsilon/k]$ can be extended into at most two functions on $[0, (i+1)\epsilon/k]$, by selecting the diagonals in $(i\epsilon/k, (i+1)\epsilon/k)$ that emanate from the right endpoint of the function.

The second property yields $M_{i+1}^1(\epsilon, \mathcal{F}_k) \leq 2^i M_1^1(\epsilon, \mathcal{F}_k) \leq 2^{i+2}k/\epsilon$.

Now consider the case $d = 2$: the domain $[0,1]^2$ is decomposed into $(k/\epsilon)^2$ equal-sized cells. Consider a function employed in $d = 1$ case along one axis, say x_1, then place a function from $d = 1$ in each plane $x_1 = i\epsilon/k$, $i = 1, 2, \ldots, k/\epsilon$. See Fig. 2 for an illustration. Then place at most two planes in each cell by choosing three points at a time.

In general, let $M_i^j(\epsilon, \mathcal{F}_k)$ denote the functions defined on the domain $[0, i\epsilon/k] \times [0,1]^{j-1}$ such that $M_{k/\epsilon}^j(\epsilon, \mathcal{F}_k) = N_\infty^j(\epsilon, \mathcal{F}_k)$, for $j = 1, 2, \ldots, d$. Then we have the following inequalities, for $j = 2, 3, \ldots, d$:

(a) $M_0^j(\epsilon, \mathcal{F}_k) = M_{k/\epsilon}^{j-1}(\epsilon, \mathcal{F}_k)$ since all functions are confined to the $(j-1)$-hyperplane; and

(b) $M_{i+1}^j(\epsilon, \mathcal{F}_k) \leq 2M_i^j(\epsilon, \mathcal{F}_k)$ since any function on $[0, i\epsilon/k] \times [0,1]^{(j-1)}$ can be extended into at most two functions on $[0, (i+1)\epsilon/k] \times [0,1]^{(j-1)}$.

Then we have the following inequality

$$M_{k/\epsilon}^d(\epsilon, \mathcal{F}_k) \leq 2^{\frac{k}{\epsilon}(\frac{k}{\epsilon}-1)} M_0^d(\epsilon, \mathcal{F}_k) \leq 2^{\frac{k}{\epsilon}(\frac{k}{\epsilon}-1)} M_{\frac{k}{\epsilon}}^{d-1}(\epsilon, \mathcal{F}_k)$$
$$\leq 2^{\frac{k}{\epsilon}(\frac{k}{\epsilon}-1)^2} M_{\frac{k}{\epsilon}}^{d-2}(\epsilon, \mathcal{F}_k) \ldots \leq 2^{\frac{k}{\epsilon}(\frac{k}{\epsilon}-1)^{d-1}} M_{\frac{k}{\epsilon}}^1(\epsilon, \mathcal{F}_k) \leq \frac{2k}{\epsilon} 2^{\frac{k}{\epsilon}((\frac{k}{\epsilon})^{d-1}+1)}$$

which proves the theorem. □

Upperbound estimates for $\log N_\infty(\epsilon, \mathcal{F}_k)$ in the form $J\epsilon^{-d/\alpha}$ with unspecified J and α were derived by Kolmogorov and Tikhomirov [11]. Similar estimates for classes of differentiable functions and sets with differentiable boundaries were obtained by Dudley ([7], chapter 7). The above estimate yields precise values of the underlying constants which are required for PAC-style results. In terms of the order, our estimate is identical with estimate based on a more restrictive property of differentiability; since neural networks considered here satisfy infinite differentiability, it is an open problem if indeed this property can be exploited to obtain a cover with lower order.

Lemma 3.4 *For the class of feedforward neural networks $\mathcal{F}_A^\gamma$ of Eq (2.2), we have*

$$N_\infty(\epsilon, \mathcal{F}_A^\gamma) \leq \frac{2\gamma A^2 l}{\epsilon} e^{\left\{\frac{\gamma A^2 l}{\epsilon}\left[\left(\frac{\gamma A^2 l}{\epsilon}-1\right)^{d-1}+1\right]\right\}}.$$

Proof: Let us expand $f_w(x)$ as $\sum_{j=1}^{l} a_j \sigma(\sum_{i=1}^{d} b_{ji}x_i + t_j)$. The estimate on the Lipschitz constant can be obtained by maximizing the partial derivative $\frac{\partial f_w}{\partial x_j}$. Since $\sigma(z) = \tanh(\gamma z)$, $\max \frac{\partial \sigma(z)}{\partial z} = \gamma$, $\frac{\partial f_w}{\partial x_j} = \sum_{i=1}^{l} a_i \sigma'(\sum_{i=1}^{d} b_{ji}x_i + t_j) b_{ij} \leq \gamma A^2 l$, and the lemma follows. □

4 Function Estimation

We first define the Lipschitz property of $\Theta(.)$ and consider its impact on the cover size for $\Theta(\mathcal{F}_A)$. Consider $\mathcal{F} = \{f : [0,1]^d \mapsto \Re\}$. Given $f_1, f_2 \in \mathcal{F}$, we say that $f_1 \leq f_2$ if $f_1(x) \leq f_2(x)$ for all $x \in [0,1]^d$. And the function $|f_1 - f_2|$ is defined as $|f_1(x) - f_2(x)|$ at every $x \in [0,1]^d$. For $g_1, g_2 \in \mathcal{G}$ the same definitions apply. The cost function $\Theta(.)$ defined on $\mathcal{F}$ satisfies *Lipschitz property* if there exists a positive constant L_Θ such that

$$|\Theta(f_1) - \Theta(f_2)| \leq L_\Theta |f_1 - f_2|$$

for all $f_1, f_2 \in \mathcal{F}$. The main impact of this property is that a cover for $\mathcal{F}_A$ can be converted into that for $\{\Theta(f_w) : w \in [-A, A]^{l(d+2)}\}$.

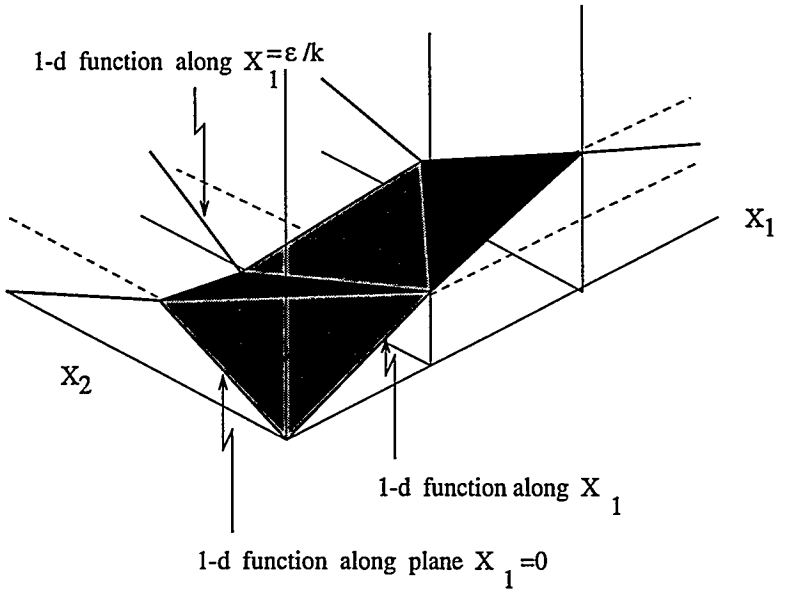

Figure 2: Illustration of cover for $d = 2$.

Lemma 4.1 *Let $\Theta(.)$ be a Lipschitz function defined on a class $\mathcal{F} = \{f : [0,1]^d \mapsto \Re\}$. Then for the class of functions $\Theta(\mathcal{F}) = \{\Theta(f) : f \in \mathcal{F}\}$ we have*

$$N_\infty(\epsilon, \Theta(\mathcal{F})) \leq N_\infty(\epsilon/L_\Theta, \mathcal{F}) \quad \textit{and} \quad N_1(\epsilon, \mu, \Theta(\mathcal{F})) \leq N_1(\epsilon/L_\Theta, \mu, \mathcal{F}).$$

Proof: We have $|\Theta(f_1) - \Theta(f_2)|(x) \leq L_\Theta |f_1(x) - f_2(x)|$ for all $x \in [0,1]^d$, which implies $\| \Theta(f_1) - \Theta(f_2) \|_\infty \leq L_\Theta \| f_1 - f_2 \|_\infty$ and $\mu|\Theta(f_1) - \Theta(f_2)| \leq L_\Theta \mu |f_1 - f_2|$. Thus an ϵ/L_Θ-cover for $\mathcal{F}$ constitutes an ϵ-cover for $\Theta(\mathcal{F})$. □

We first consider the sample size based on the cover of Section 3.1 for neural networks.

Theorem 4.1 *Consider the class of feedforward neural networks of Eq (2.1)*

$$\mathcal{F}_\mathcal{A} = \left\{ f_w(x) = \sum_{i=1}^{l} a_i \sigma(b_i^T x + t_i) : a_i, t_i \in [-A, A], b_i \in [-A, A]^d \right\},$$

and assume that the cost function $\Theta(.)$ is Lipschitz with constant L_Θ. Given a sample of size at least

$$\frac{256 l^2 A^2}{\epsilon^2} \max \left\{ \ln\left(\frac{8}{\delta}\right), \frac{(d+2)\epsilon^2}{256 l A^2}, (2l(d+2)+2)\log 5 + 2l(d+2)\log\left(\frac{16 l A L_\Theta}{\epsilon}\right) \right\}$$

the empirically best neural network $\hat{f}_w$ in $\mathcal{F}_\mathcal{A}$ approximates the best expected f_w^* in $\mathcal{F}_\mathcal{A}$ such that

$$P\left[I(\hat{f}_w) - I(f_w^*) > \epsilon \right] < \delta.$$

Proof: The sample size estimate is based on utilizing the covering number of Lemma 3.2. By the result of Vapnik [24] the condition $P[I(\hat{f}_w) - I(f_w^*) > \epsilon] < \delta$ is implied by $P\left[\sup_{f_w \in \mathcal{F}_\mathcal{A}} |\hat{I}(f_w) - I(f_w)| > \epsilon/2 \right] < \delta$. By noting that $f_w(x) \leq lA$ for all $w \in [-A, A]^{l(d+2)}$ and $x \in [0,1]^d$, from Pollard [16], we have

$$P\left[\sup_{f_w \in \mathcal{F}_\mathcal{A}} |\hat{I}(f_w) - I(f_w)| > \epsilon \right] \leq 8 e^{\frac{-n\epsilon^2}{256 l^2 A^2}} + P\left[\log N_1(\epsilon/16, P_n, \Theta(\mathcal{F}) \geq \frac{n\epsilon^2}{256 l^2 A^2} \right].$$

By Lemma 3.2 and 4.1, we have

$$N_1(\epsilon/16, P_n, \Theta(\mathcal{F}_W)) \leq E \epsilon^{-2l(d+2)}$$

where $E = \max(C_W^2, k_0)(16lAL_\Theta)^{2l(d+2)}$. The second term in the above equation can be made zero by choosing n such that

$$2^{\frac{n\epsilon^2}{256l^2A^2}} \geq E\epsilon^{-2l(d+2)}$$

or equivalently

$$n \geq \frac{256l^2A^2}{\epsilon^2}(\ln E + 2l(d+2)\log(1/\epsilon)).$$

From Vapnik [24] we have $C_W = \frac{1.5}{(l(d+2)-1)!}$ for $n > l(d+2)$. Now we can show that for $k \geq 5^{2l(d+2)+2}$ we have the condition $(1+4\log k)^{l(d+2)} \leq \sqrt{k}$ satisfied. To see this notice first that this condition is satisfied if $(5\log k)^{l(d+2)} \leq \sqrt{k}$ for $k > 2$; this condition is in turn satisfied if $5^{l(d+2)}k^{1/x} \leq k^{1/2}$ since $\log y \leq y^{1/z}$ where z is a finite integer. The bound then follows by choosing $x = 3$. Using this value in place of k_0 we obtain

$$\log E = (2l(d+2)+2)\log 5 + 2l(d+2)\log(16lAL_\Theta).$$

Then the condition $8e^{\frac{-n\epsilon^2}{256l^2A^2}} = \delta$ is ensured by choosing $n \geq \frac{256l^2A^2}{\epsilon^2}\ln(8/\delta)$. □

Remark: By using the result of Lugosi and Zeger [12], we can show that

$$P[I(\hat{f}) - I(f^*) > \epsilon] \leq 4\left(\frac{128e(l+1)lAL_\Theta}{\epsilon}\right)^{l(2d+3)+1} e^{\frac{-n\epsilon^2}{2^5l^2A^2L_\Theta^2}}$$

which yields the following sample size

$$\frac{2^5l^2A^2L_\Theta^2}{\epsilon^2}\left(\ln(4/\delta) + (l(2d+3)+1)\ln\left(\frac{128e(l+1)lAL_\Theta}{\epsilon}\right)\right).$$

We now consider the sample size for empirical estimation based on the results of Section 3.2.

Theorem 4.2 *Consider the class of feedforward neural networks of Eq (2.2)*

$$\mathcal{F}_\mathcal{A}^\gamma = \left\{\sum_{i=1}^{l} a_i\sigma(b_i^Tx + t_i) : a_i, t_i \in [-A, A], b_i \in [-A, A]^d, \sigma(z) = \tanh(\gamma z), 0 < \gamma < \infty\right\},$$

and assume that the cost function $\Theta(.)$ is Lipschitz with constant L_Θ. Given a sample of size at least

$$\frac{128l^2A^2L_\Theta^2}{\epsilon^2}\left[\ln\left(\frac{2\gamma A^3l^2L_\Theta^2}{\epsilon\delta}\right) + 4\ln\left(\frac{8lAL_\Theta}{\epsilon}\right) + \frac{\gamma A^3l^2L_\Theta^2}{\epsilon}\left(\left(\frac{\gamma A^3l^2L_\Theta^2}{\epsilon} - 1\right)^{d-1} + 1\right)\right],$$

the empirically best neural network $\hat{f}_w$ in $\mathcal{F}_\mathcal{A}^\gamma$ approximates the best expected f_w^* in $\mathcal{F}_\mathcal{A}^\gamma$ such that

$$P_X^n[I(\hat{f}_w) - I(f_w^*) > \epsilon] < \delta.$$

Proof: The sample size estimate is based on utilizing the covering number $N_\infty(\epsilon, \mathcal{F}_\mathcal{A}^\gamma)$ of Lemma 3.4. First note by Lemma 4.1 that

$$N_\infty(\epsilon, \Theta(\mathcal{F}_\mathcal{A}^\gamma)) = \frac{2\gamma A^2lL_\Theta}{\epsilon}e^{\left\{\frac{\gamma A^2lL_\Theta}{\epsilon}\left[\left(\frac{\gamma A^2lL_\Theta}{\epsilon}-1\right)^{d-1}+1\right]\right\}}.$$

From Vapnik [24] (page 190), we have

$$P\left[\sup_{f_w \in \mathcal{F}_\mathcal{A}^\gamma} |\hat{I}(f_w) - I(f_w)| > lAL_\Theta\epsilon\right] \leq 18N_\infty(\epsilon, \Theta(\mathcal{F}_\mathcal{A}^\gamma))\, ne^{\epsilon^2n/4}$$

where $\Theta(f_w)(x) \leq L_\Theta lA$ for all $x \in [0,1]^d$. Thus we have

$$P\left[I(\hat{f}_w) - I(f_w^*) > \epsilon\right] \leq 18N_\infty\left(\frac{\epsilon}{2lAL_\Theta}, \Theta(\mathcal{F}_A)\right) ne^{\frac{-\epsilon^2n}{(8lAL_\Theta)^2}}.$$

For neural networks, we have

$$N_\infty\left(\frac{\epsilon}{2la}, \Theta(\mathcal{F}_\mathcal{A}^\gamma)\right) = \frac{2\gamma A^3 l^2 L_\Theta^2}{\epsilon} e^{\left\{\frac{\gamma A^3 l^2 L_\Theta^2}{\epsilon}\left[\left(\frac{\gamma A^3 l^2 L_\Theta^2}{\epsilon}-1\right)^{d-1}+1\right]\right\}}.$$

The sample size is obtained by noting that the general form $\delta \geq ane^{-bn}$ is ensured by choosing $n \geq \frac{2}{b}\ln(a/b^2\delta)$ as follows. First note that above condition is implied by

$$n \geq 1/b\ln(\frac{a}{b^2\delta}) + 1/b\ln(b^2 n).$$

Since $\ln x \leq \sqrt{x}$, this condition is ensured by $n \geq 1/b\ln(\frac{a}{b^2\delta}) + \sqrt{n}$. Now the condition $n \geq \sqrt{n}+c$ is ensured by choosing $n \geq 2c$ for $c \geq 2$. To see this first note that $n - \sqrt{n}$ is an increasing function, and hence if $n_0 \geq \sqrt{n_0}+c$ for some n_0 then $n \geq \sqrt{n}+c$ for all $n \geq n_0$. Let $n_0 = 2c$, then $n_0 > c+\sqrt{2c}$ for $c \geq 2$; thus $n_0 > \sqrt{n_0}+c$. Thus for the general form we have $n = \frac{2}{b}\ln(\frac{a}{b^2\delta})$ and the theorem follows by substituting the appropriate quantities. □

References

[1] N. Alon, S. Ben-David, N. Cesa-Bianchi, and D. Hausler. Scale-sensitive dimensions, uniform convergence, and learnability. In *Proc. of 1993 IEEE Symp. on Foundations of Computer Science*, 1993.

[2] M. Anthony and P. Bartlett. Function learning from interpolation. NeuroCOLT Technical Report Series NC-TR-94-013, Royal Holloway, University of London, 1994.

[3] K. Apsitis, R. Freivalds, and C. H. Smith. On the inductive inference of real valued functions. In *Proc. of 8th Ann. ACM Conf. on Computational Learning Theory*, 1995.

[4] P. Auer, P. M. Long, W. Mass, and G. J. Woeginger. On the complexity of function learning. In *Proc. of 6th Ann. ACM Conf. on Computational Learning Theory*, pages 392–401, 1993.

[5] P. L. Bartlett, P. M. Long, and R. C. Williamson. Fat-shattering and the learnability of real-values fucntions. In *Proc. of 7th Ann. ACM Conf. on Computational Learning Theory*, 1994.

[6] S. Ben-David, N. Cesa-Bianchi, D. Haussler, and P. Long. Characterizations of learnability for classes $\{0,\ldots,n\}$-valued functions. In *Proc. of 5th Ann. ACM Conf. on Computational Learning Theory*, 1992.

[7] R. Dudley. *École d'Été de Probabilités de St. Flour 1982*, volume 1097 of *Lecture Notes in Mathematics*, chapter A Course on Empirical Processes, pages 2–142. Springer-Verlag, New York, 1984.

[8] D. Haussler. Decision theoretic generalizations of the PAC model for neural net and other learning applications. *Information and Computation*, 100:78–150, 1992.

[9] D. Haussler. Sphere packing numbers for subsets of boolean n-cube with bounded Vapnik-Chervonenkis dimension. *Journal of Combinatorial Theory, Series A*, 69(2):217, 1995.

[10] D. Kimber and P. M. Long. The learning complexity of smooth functions of a single variable. In *Proc. of the 1992 Workshop on Computational Learning*, pages 153–159, 1992.

[11] A. N. Kolmogorov and V. M. Tikhomirov. ϵ-entropy and ϵ-capacity of sets in function spaces. *American Mathematical Society Translations: Series 2*, 17:277–364, 1961.

[12] G. Lugosi and K. Zeger. Nonparametric estimation via empirical risk minimization. *IEEE Transactions on Information Theory*, 41(3):677–687, 1995.

[13] M. MacIntyre and E. D. Sontag. Finiteness results for sigmoidal neural networks. In *Proceedings of the 25th Annual ACM Symposium on the Theory of Computing*, pages 325–334. 1993.

[14] B. K. Natarajan. *Machine Learning: A Theoretical Approach.* Morgan Kaufmann Pub. Inc., San Mateo, California, 1991.

[15] D. Nolan and D. Pollard. U-Processes: Rates of convergence. *Annals of Statistics*, 15(2):780–799, 1987.

[16] D. Pollard. *Convergence of Stochastic Processes.* Springer-Verlag, New York, 1984.

[17] B. L. S. Prakasa Rao. *Nonparametric Functional Estimation.* Academic Press, New York, 1983.

[18] N. S. V. Rao and V. Protopopescu. Algorithms for PAC learning of functions with smoothness properties. In *Proceedings of the Fourth International Symposium on Artificial Intelligence and Mathematics*, pages 134–137, 1996.

[19] V. Roychowdhury, K. Siu, and A. Orlitsky, editors. *Theoretical Advances in Neural Computation and Learning.* Kluwer Academmic Pub., 1994.

[20] J. Shawe-Taylor. Sample sizes for sigmoidal neural networks. In *Proceedings of the 8th Annual Conference on Computational Learning Theory*, pages 258–264, 1995.

[21] H. U. Simon. Bounds on the number of examples needed for learning functions. In J. Shawe-Taylor and M. Anthony, editors, *Computational Learning Theory: EUROCOLT'93.* Oxford University Press, 1994.

[22] J. F. Traub, G. W. Wasilkowski, and H. Woziniakowski. *Information-Based Complexity.* Academic Press, Inc., 1988.

[23] L. G. Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.

[24] V. Vapnik. *Estimation of Dependences Based on Empirical Data.* Springer-Verlag, New York, 1982.

[25] V. N. Vapnik. *The Nature of Statistical Learning Theory.* Springer-Verlag, New York, 1995.

[26] H. White. Connectionist nonparametric regression:Multilayer feedforward networks can learn arbitrary mappings. *Neural Networks*, 3:535–549, 1990.

[27] H. White. Nonparametric estimation of conditional quantiles using neural networks. In C. Page and R. Le Page, editors, *Computing Science and Statistics.* Springer Verlag, 1992.