ORNL--96009545

# On PAC Learning of Functions with Smoothness Properties Using Feedforward Sigmoidal Networks

Nageswara S. V. Rao and Vladimir A. Protopopescu
Center for Engineering Systems Advanced Research
P. O. Box 2008
Oak Ridge National Laboratory
Oak Ridge, Tennessee 37831-6364
{raons,protopopesva}@ornl.gov

**DISCLAIMER**

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

Paper submitted to *Proceedings of IEEE* journal.

Can't conf.
info incomplete

†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-AC05-84OR21400 with Martin Marietta Energy Systems, Inc.

# On PAC Learning of Functions with Smoothness Properties Using Feedforward Sigmoidal Networks

Nageswara S. V. Rao and Vladimir A. Protopopescu
Center for Engineering Systems Advanced Research
P. O. Box 2008
Oak Ridge National Laboratory
Oak Ridge, Tennessee 37831-6364
{raons,protopopesva}@ornl.gov

### Abstract

We consider *Probably and Approximately Correct* (PAC) learning of an unknown function $f : [0,1]^d \mapsto [0,1]$, based on finite samples using feedforward sigmoidal networks. The unknown function $f$ is chosen from the family $\mathcal{F} \cap \mathcal{C}([0,1]^d)$ or $\mathcal{F} \cap \mathcal{L}^\infty([0,1]^d)$, where $\mathcal{F}$ has either bounded modulus of smoothness or bounded capacity or both. The learning sample is given by $(X_1, f(X_1))$, $(X_2, f(X_2))$, $\cdots$, $(X_n, f(X_n))$, where $X_1, X_2, \cdots, X_n$ are independently and identically distributed (iid) according to an unknown distribution $P_X$. For simplicity, we consider the feedforward networks with a single hidden layer of $1/(1+e^{-\gamma z})$-units and bounded parameters, but the results can be extended to other neural networks where the hidden units satisfy suitable smoothness conditions. We analyze three function estimators based on: (i) nearest neighbor rule, (ii) local averaging, and (iii) Nadaraya-Watson estimator, all computed using the Haar system. It is shown that given a sufficiently large sample, each of these estimators approximates the best neural network to any given error with arbitrarily high probability. This result is crucial for establishing the essentially equivalent capabilities of neural networks and the above estimators for PAC learning from finite samples. The *practical* importance of this "equivalence" stems from the fact that computing a neural network which approximates (in the above sense) the best possible one is computationally difficult, whereas the three estimators above are linear-time computable in terms of the sample size.

## 1 Introduction

Retracing the modern history of research on machine learning, Vapnik [24] identified four major turning points: (i) the first learning machines based on Rosenblatt's perceptron (the 60's); (ii) the foundations of the theory based on Vapnik-Chervonenkis' and Chaitin's results (the 70's); (iii) the neural networks tide (the 80's); and (iv) the return to the origin including alternatives to neural networks and refocussing on small (vs. asymptotic) samples (the 90's). The search for alternatives has been prompted by a series of drawbacks of the neural network approach to machine learning and, in particular, to learning of functions. Indeed, despite notable successes in specific applications, general learning theory has not been advanced by neural networks and the performance analysis of neural network learning methods is still under development — with many issues unresolved. In particular, when the sample is finite (and small) most neural network learning algorithms are either not proven to converge or, when convergent to a local minimum, shown to perform poorly. From a practical viewpoint, fitting a neural network to data is still an artful, non-systematic, time-consuming, and often frustrating operation, while the complexity of computing a neural network remains high. Due to their wide use for learning and approximation [19] it is therefore crucial to better understand and improve upon the performance of neural network algorithms. In this paper, we address two aspects of neural network algorithms for function learning, namely: (i) learning from finite samples and (ii) efficient computational alternatives.

We address these issues in a statistical formulation along the lines of Cheng and Titterington [5], in particular in the framework of *Probably and Approximately Correct* (PAC) of Valiant [21]. Given independently and identically distributed (iid) points and the values of an unknown function chosen from a family $\mathcal{F}$, we consider the function estimation using feedforward neural networks. The unknown function is assumed to be continuous or essentially bounded with a bounded modulus of smoothness. We consider feedforward networks with a single hidden layer of $1/(1+e^{-\gamma z})$-units and bounded weights. The results can be extended to neural networks with more hidden layers if units satisfy suitable smoothness conditions.

We provide PAC results for three function estimators based on: (i) the nearest neighbor rule, (ii) local averaging, and (iii) the Nadaraya-Watson estimator, all computed using the Haar function system. We then show that for a sufficiently large sample, each of the above estimators approximates the "best" neural network to any given error, with any desired probability.

Another aspect — perhaps a more important one from a practical view point — is the lack of algorithms with efficient finite sample performance to train a neural network. Experimentation with several widely employed gradient search methods, e. g. backpropagation, do not seem to yield very good rate of convergence in a number of applications. The existing guarantees of asymptotic convergence or finite sample results are conditioned on a number of smoothness and/or martingale conditions (White [25], Nedeljkovic [12], Rao *et al.* [18]) that are very difficult to verify in practical cases. To overcome this difficulty, we specialize the three estimators above by employing Haar kernels. Our choice of Haar functions based estimators is dictated by their computational convenience and their ability to yield finite sample results. As a result, the estimated function value at a given point can be computed in $O(n)$ time, for all three cases. With preprocessing, the second and third can be computed in $O((\log n)^d)$ time using a range-tree precomputed in $O(dn(\log n)^d)$ time. The practical implications of the equivalence of these estimators to neural networks could hardly be overestimated since these estimators are linear-time computable in terms of the sample size.

Apart from providing computationally efficient approximations to neural networks, our finite sample results for the three estimators could be of independent interest as function estimators. Recent advances in PAC estimation of functions established that a function that achieves small empirical error on an iid sample yields a PAC approximation, under the finiteness of a combinatorial parameter such as the fat-shattering index [3, 2]. Our results differ from those based on capacity (or related combinatorial parameters) in two directions: (a) under mild smoothness conditions on the function and/or the density, we can obtain stronger guarantees for the error between the function and the estimator; and (b) our estimators can be computed in *linear time*, unlike the general PAC solutions that usually require solving NP-hard problems. Despite their long history, we are unaware of any previous finite sample results for the above three estimators. Computationally our methods are similar to the regression tree [4], but their results are only asymptotic.

The paper is organized as follows. Preliminaries and background material are presented in Section 2. The finite sample results for the three estimators are presented in Section 3. The equivalence results between these estimators and feedforward neural networks is discussed in Section 4.

## 2 Preliminaries

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

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

where $a = (a_1, a_2, \ldots, a_l) \in \Re^l$ and $w$ is the *weight vector* of the network consisting of $a, b_1, b_2, \ldots, b_l$ and $t_1, t_2, \ldots, t_l$. We consider neural networks with bounded weights such that $w \in [-W, +W]^{l(d+2)}$ for some fixed positive $W < \infty$. We consider hidden units of the particular form $\sigma(z) = 1/(1+e^{-\gamma z})$, for $\gamma, z \in \Re$. Let $\mathcal{F}_{\mathcal{W}}$ denote set of all functions implemented by neural networks of the above kind, with fixed $d$ and $m$, and various values of $w$.

A *training n-sample* of a function $f : [0,1]^d \mapsto [0,1]$, chosen from a family $\mathcal{F}$, is given by $(X_1, f(X_1)), (X_2, f(X_2)), \ldots, (X_n, f(X_n))$ where $X_1, X_2, \ldots, X_n$, $X_i \in [0,1]^d$, are iid according to an *unknown* distribution $P_X$ ($X = [0,1]^d$). The *function learning problem* is to estimate a function $g : [0,1]^d \mapsto [0,1]$, based on the sample, such that the expected error defined as

$$I(g) = \int |g(X) - f(X))| dP_X \tag{2.1}$$

is minimized over the class $\mathcal{G}$ of estimators. Consider $\mathcal{G} = \mathcal{F}_{\mathcal{W}}$ and let $f_w^* \in \mathcal{F}_{\mathcal{W}}$ minimize $I(.)$, i. e., $f_w^*$ is the best possible neural network in the sense of (2.1). In general, $f_w^*$ cannot be computed since both $f$ and $P_X$ are unknown. Furthermore, since no restrictions are placed on $P_X$, it is not always possible to infer $f_w^*$ (with probability one) based on a finite sample. Consequently, most often only an approximation $g$ to $f_w^*$ is feasible. We consider conditions under which an approximation $g$ to $f_w^*$ satisfies

$$P[I(g) - I(f_w^*) > \epsilon] < \delta \tag{2.2}$$

for arbitrarily specified $\epsilon$ and $\delta$, $0 < \epsilon, \delta < 1$, where $P = P_X^n$ is the product measure on the set of all iid $n$-samples. Thus the approximation "error" of $g$ is to be bounded by $\epsilon$ with a probability of $1 - \delta$ (given a sufficiently large sample).

Let $\hat{f}_w$ minimize the empirical risk function $I_{emp}(g) = \frac{1}{n} \sum_{i=1}^{n} |g(x_i) - f(x_i)|$ over all $g \in \mathcal{F}_{\mathcal{W}}$. By making use of the bounded capacity of $\mathcal{F}_{\mathcal{W}}$ (or using other characterizations, see Anthony [1], Haussler [9]), one can show that $\hat{f}_w$ satisfies the condition (2.2) given sufficiently large sample. However, the computational problem of obtaining $\hat{f}_{emp}$ by using neural networks is riddled with difficulties. In practice the training algorithm is terminated after a certain number of iterations which results only in a (sometimes poor) approximation of $\hat{f}_{emp}$. We show here that both the nearest neighbor rule, $\bar{f}$, and the regressogram, $\hat{f}$, provide such approximations. In particular, we show that, given sufficiently large sample, we have $P[I(g) - I(f_w^*) > \epsilon] < \delta$ for both $g = \bar{f}$ and $g = \hat{f}$.

We consider another cost functional defined by

$$I_\infty(g) = \sup_{x \in [0,1]^d} |g(x) - f(x)| \tag{2.3}$$

and show that the Nadaraya-Watson estimator , $\tilde{f}$, PAC approximates, along the lines of Eq. (2.2), to the best possible neural network under this cost functional.

Let $Q = [0,1]^d$, and let $\mathcal{C}(Q)$ and $\mathcal{L}^\infty(Q)$ denote the classes of continuous and essentially bounded functions defined on $Q$, respectively. For $f \in L^\infty(Q)$, we have

$$\| f \|_\infty = \operatorname{ess\,sup}\{|f(x)| : x \in Q\}.$$

The modulus of smoothness of $f \in L^\infty(Q)$ is defined as

$$\omega_\infty(f;r) = \sup_{|h|_\infty < r} (\operatorname*{ess\,sup}_{Q(h)} |f(x+h) - f(x)|)$$

where $Q(h) = \{x \in Q : x + h \in Q\}$ and $|h|_\infty = \max(|h_1|,\ldots,|h_d|)$. We note that for continuous functions, $f \in \mathcal{C}(Q)$, the modulus of smoothness coincides with the ordinary modulus of continuity defined as

$$\omega_\infty(f;r) = \sup_{|x-y|_\infty < r,\ x,y \in Q} |f(x) - f(y)|.$$

For a family $\{A_\gamma\}_{\gamma\in\Gamma}$, $A_\gamma \subseteq A$, and for a finite set $\{a_1, a_2, \ldots, a_n\} \subseteq A$, we have [22]:

$$\Pi_{\{A_\gamma\}}(\{a_1,a_2,\ldots,a_n\}) = \{\{a_1,a_2,\ldots,a_n\} \cap A_\gamma\}_{\gamma\in\Gamma},$$

$$\Pi_{\{A_\gamma\}}(n) = \max_{a_1,a_2,\ldots,a_n} |\Pi_{\{A_\gamma\}}(\{a_1,a_2,\ldots,a_n\})|.$$

The following identity is established in [22]: $\Pi_{\{A_\gamma\}}(n) = \begin{cases} 2^n & \text{if } n \le k \\ < 1.5\frac{n^k}{k!} & \text{if } n > k. \end{cases}$

Notice that for a fixed $k$, the right hand side increases exponentially with $n$ until it reaches $k$ and then varies as a polynomial in $n$ with fixed power $k$. This quantity $k$ is called the *Vapnik-Chervonenkis* (VC) dimension of the family of sets $A_\gamma$.

For a set of functions, the *capacity* [23] is defined as the largest number $h$ of pairs $(x_i, y_i)$ that can be subdivided in all possible ways into two classes by means of rules of the form $\{\Theta[(y - f(x))^2 + \beta]\}_{(f,\beta)}$ where $(f,\beta) \in \mathcal{F} \times \Re$ and $\Theta(z)$ is the Heaviside step-function defined as

$$\Theta(z) = \begin{cases} 1 & \text{if } z \ge 0 \\ 0 & \text{f } z < 0. \end{cases}$$

Thus the capacity of a family of functions $\mathcal{F}$ is the VC dimension of the set of indicator functions $\{\Theta[(y - f(x))^2 + \beta]\}_{(f,\beta)\in\mathcal{F}\times\Re}$.

For $m = 0,1,\ldots$, let $Q_m$ denote a family of diadic cubes such that $[0,1]^d = \bigcup_{J\in Q_m} J$, $J \cap J' = \emptyset$ for $J \ne J'$, and the $d$-dimensional volume of $J$, denoted by $|J|$, is $2^{-dm}$. Let $1_J(x)$ denote the indicator function of $J \in Q_m$: $1_J(x) = 1$ if $x \in J$, and $1_J(x) = 0$ otherwise. For given $m$, we define $P_m : \mathcal{L}^\infty(Q) \mapsto \mathcal{L}^\infty(Q)$ by

$$P_m f(x) = \frac{1}{|J|}\int_J f(y)dy$$

for $x \in J$ and $J \in Q_m$ [6]. Consider a kernel given by $P_m(x,y) = 2^{dm} \sum_{J\in Q_m} 1_J(x)1_J(y)$ for $x, y \in Q$.

Then an estimator for a density $p \in \mathcal{L}^\infty(Q)$ based on $n$-sample is given by (Ciesielski [6])

$$\tilde{p}_{m,n} = \frac{1}{n}\sum_{j=1}^n P_m(x, X_j)$$

which can also be written in the form $\tilde{p}_{m.n} = \sum_{J\in Q_m} n(J)h_J(x)$ with $n(J) = \frac{1}{n}|\{j : X_j \in J\}|$ and $h_J(x) = \frac{1}{|J|}1_j(x)$.

The following result, due to Ciesielski [6], will be used in the construction of the PAC learning algorithms.

**Lemma 2.1** *[6] Let $0 < \alpha \leq 1$ and $f \in \mathcal{C}(Q)$ or $f \in \mathcal{L}^\infty(Q)$ be given. Then the condition $\omega(f;r) = O(r^\alpha)$ as $r \to 0_+$ implies*

$$\| f - P_m f \|_\infty = C/2^{\alpha m} \quad as \quad m \to \infty$$

*for some $C > 0$.*

## 3 Function Estimation: Finite Sample Results

The unknown function $f$ is chosen from the family [1] $\mathcal{F} \cap \mathcal{C}([0,1]^d)$ or $\mathcal{F} \cap \mathcal{L}^\infty([0,1]^d)$, where $\mathcal{F}$ has either bounded modulus of smoothness or bounded capacity or both. We consider three types of function estimators for class of continuous functions (the case of essentially bounded functions follows directly). The first one is based on the nearest neighbor rule applied to each cell of $Q_m$. The second is a local estimator based on "averaging" the function values within each cell of suitably chosen $Q_m$. When the functions have bounded moduli of smoothness, these estimators provide *distribution-free* results. The third estimator, called Nadaraya-Watson, applies to a more particular case where the density exists and also satisfies some smoothness properties. Not surprisingly, the Nadaraya-Watson estimator based on the Haar system provides better guarantees under the cost $I_\infty(.)$.

### 3.1 Local Nearest Neighbor Rule

Based on the $n$-sample $(X_1, f(X_1)), (X_2, f(X_2)), \ldots, (X_n, f(X_n))$, the nearest neighbor estimator for the function $f$ is given by

$$\bar{f}_{m,n}(x) = \sum_{J \in B_m} 1_J(x) \mathcal{N}_J(x) \tag{3.1}$$

where for $x \in J$, $\mathcal{N}_J(x)$ yields $f(X_i)$ such that $X_i \in J$ is closest to $x$ in sup norm. As shown later, this estimator has a higher computational complexity than the other two when preprocessing is not allowed and also provides a weaker performance guarantee. The trade-off is that it only requires a bounded modulus of smoothness and does *not require bounded capacity.*

**Theorem 3.1** *Consider a family of continuous functions $\mathcal{F} \subseteq \mathcal{C}([0,1]^d)$, with range $[0,1]$ such that for every $f \in \mathcal{F}$, we have $\omega_\infty(f;r) \leq kr$ as $r \to 0$. Suppose that the size of the sample, $n$, is larger than*

$$\frac{2^{2md+4}k^2d^3e}{\epsilon^2} \ln^2\left(\frac{2^{3md+4}k^2d^3e}{\delta\epsilon^2}\right) + 2$$

*where $m = \log(2C/\epsilon)$ is larger than a suitable constant $m_0$. Then for $X$ distributed according to $P_X$ and any $f \in \mathcal{F}$, we have $P\left[E|f(X) - \bar{f}_{m,n}(X)| > \epsilon\right] < \delta$.*

**Proof:** First we consider

$$P[E|f(X) - \bar{f}_{m,n}(X)| > \epsilon] \leq P[E|f(X) - P_m f(X)| > \epsilon/2] + P[E|P_m f(X) - \bar{f}_{m,n}(X)| > \epsilon/2].$$

Under the hypothesis $\omega_\infty(f;r) \leq kr$ we have $\| f - P_m f \|_\infty \leq C/2^m$ due to Lemma 2.1 for $m \geq m_0$, for suitable constants $m_0$ and $C$. By choosing $m$ such that $C/2^m \leq \epsilon/2$, the first term on the right hand side becomes zero. The second term is upperbounded by

$$2^{md} P\left[\sup_{J \in B_m} E(|P_m f(X) - \bar{f}_{m,n}(X)| 1_J(X)) > \epsilon/2^{md+1}\right].$$

[1] Some additional measurability conditions are required on $\mathcal{F}$, which are assumed to be satisfied throughout the paper, and are not repeated here since they do not play a direct role in our proofs; see Pollard [13] for details.

Now we note that $f \in \mathcal{F}$ has bounded modulus of continuity in each $J \in B_m$ with a constant $k$. By the mean value theorem, $|\frac{1}{|J|}\int_J f(x)dx - f(x)| \leq k/2^m$ for $x \in J$, and

$$\min_{x \in J} f(x) \leq \frac{1}{|J|}\int_J f(x)dx \leq \max_{x\in J} f(x).$$

From Rao [17], we have the following result: let $\mathcal{F}_{L(k)}$ denote the set of functions $f : [0,1]^d \mapsto [0,1]$ that are Lipschitz with constant $k$, i.e. for every $f \in \mathcal{F}_{L(k)}$, we have $|f(x) - f(y)| \leq k|x-y|_\infty$ for all $x, y \in [0,1]^d$. Given a sample of size at least

$$\frac{4k^2d^3e}{\epsilon^2}\ln^2\left(\frac{4k^2d^3e}{\delta\epsilon^2}\right) + 2,$$

we have $P\left[\sup_{f\in\mathcal{F}_{L(k)}} E|f_{NN} - f| > \epsilon\right] < \delta$, where $f_{NN}$ is the nearest neighbor rule. By suitably applying this result to each individual $J \in B_m$ such that $f$ is replaced by $\int_J f(x)dx$ due to the mean value theorem above, we obtain (after a lengthy but straightforward calculation)

$$2^m P\left[\sup_{J\in B_m} E(|P_m f(X) - \bar{f}_{m,n}(X)|1_J(X)) > \epsilon/2^{m+1}\right] \leq \frac{2^{3md+4}k^2d^3e}{\epsilon^2}e^{\frac{-\sqrt{n-2}\epsilon}{2^{md+2}kd^{3/2}e^{1/2}}},$$

which yields the required bound on the sample size. □

### 3.2 Local Averaging

Based on the $n$-sample, the first estimator of the function $f$ is given by

$$\hat{f}_{m,n}(x) = \frac{1}{n}\sum_{j=1}^{n} f(X_j)P_m(x, X_j), \tag{3.2}$$

which evaluates to $\frac{1}{n}\sum_{X_j\in J}\frac{f(X_j)}{|J|}$, for $x \in J$. Estimators of this general structure are called regressograms [16] (this estimator, however, is not identical to the traditional regressogram). The next theorem provides a finite sample result for this estimate.

**Theorem 3.2** *Consider a family of continuous functions $\mathcal{F} \subseteq C([0,1]^d)$, with range $[0,1]$ and capacity $h$ such that for every $f \in \mathcal{F}$, we have $\omega_\infty(f;r) = O(r^\alpha)$ as $r \to 0$, for $0 < \alpha \leq 1$. Suppose that the size of the sample, $n$, is larger than*

$$\max\left[2\left(\frac{2^{h+1}9}{h!\delta\epsilon}\right)^{1/h}, \left(\frac{h^2 2^{4m+12}}{\epsilon^4}\right)\right]$$

*where $m = \frac{1}{\alpha}\log(3C/\epsilon)$ is larger than a suitable constant $m_0$. Then for any $X$ chosen according to the distribution $P_X$ and any $f \in \mathcal{F}$, we have $P\left[|f(X) - \hat{f}_{m,n}(X)| > \epsilon\right] < \delta$.*

**Proof:** We have for any $X$ chosen according to the distribution $P_X$,

$$\begin{aligned} P\left[|f(X) - \hat{f}_{m,n}(X)| > \epsilon\right] &\leq P[|f(X) - P_m f(X)| > \epsilon/3] + P[|P_m f(X) - Ef(X)| > \epsilon/3] \\ &\quad + P\left[|Ef(X) - \hat{f}_{m,n}| > \epsilon/3\right]. \end{aligned} \tag{3.3}$$

Under the hypothesis $\omega_\infty(f;r) \leq O(r^\alpha)$ we have $\| f - P_m f \|_\infty \leq C/2^m$ due to Lemma 2.1 for $m \geq m_0$, for suitable constants $m_0$ and $C$. By choosing $m$ such that $C/2^{\alpha m} \leq \epsilon/3$, the first term on the right hand side becomes zero.

By Chebyshev's inequality the second term is upperbounded as follows

$$3E|P_m f(X) - Ef(X)|/\epsilon \leq \frac{3}{\epsilon} \| f - P_m f \|_\infty \leq \frac{3C}{\epsilon 2^{\alpha m}}.$$

The third term is upperbounded as follows

$$\begin{aligned} P\left[|Ef(X) - \hat{f}_{m,n}| > \epsilon/3\right] &\leq P\left[\sum_{J \in B_m} \left(|P_m f(X) - \hat{f}_{m,n}(X)| 1_J(X)\right) > \epsilon/4\right] \\ &\leq 2^m P\left[\sup_{J \in B_m} \left(|P_m f(X) - \hat{f}_{m,n}(X)| 1_J(X)\right) > \epsilon/2^{m+2}\right]. \end{aligned}$$

Let $\mathcal{F}_J = \{f(x)1_J(x) | f \in \mathcal{F}\}$. The capacity of $\mathcal{F}_J$ is upperbounded by that of $\mathcal{F}$. By Vapnik's result ([22], pp. 190), we have for any $f(.)1_J(.) \in \mathcal{F}_J$,

$$P\left[\sup_{f \in \mathcal{F}} \left| \frac{1}{|J|} \int_J f(X) dP_X - \frac{1}{n} \sum_{X_i \in J} \frac{f(X_i)}{|J|} \right| > \epsilon\right] < 9\frac{(2n)^h}{h!} e^{-n\epsilon^2/4}.$$

Combining the results we have $P\left[|f(X) - \hat{f}_{m,n}(X)| > \epsilon\right] \leq \frac{2C}{\epsilon 2^{\alpha m}} + 2^{m+h} 9 \frac{n^h}{h!} e^{\frac{-n\epsilon^2}{2^{m+5}}}$. The first term is bounded by $\delta/2$ if $m = \frac{1}{\alpha}\log(4C/\epsilon\delta)$ is larger than a suitable $m_0$. This condition also ensures that $C/2^{\alpha m} \leq \epsilon/3$ which in turn assures that the first term in Eq. (3.3) is zero. The second term is bounded by $\delta/2$ as follows. Consider that $\delta = an^b e^{-nc}$ for suitable values of $a$, $b$ and $c$. First, we obtain $n = \frac{1}{c}[\ln(2/\delta) + b \ln n]$. By choosing $n \geq (2a/\delta)^{1/b}$, the required sample size is given by $n \geq 2/cb \ln n$. This latter condition can be ensured by $n \geq 4/c^2 b^2$. Thus the required value of $\delta/2$ can be ensured under the sample size $n = \max\left((2a/\delta)^{1/b}, 4b^2/c^2\right)$. The theorem follows. □

### 3.3 Nadaraya-Watson Estimator

We now present a third type of estimators that provide a better guarantee under additional conditions on the densities (similar in some sense to [20]). Based on the $n$-sample, the estimator is defined by

$$\tilde{f}_{m,n}(x) = \frac{\sum_{j=1}^{n} f(X_j) P_m(x, X_j)}{\sum_{j=1}^{n} P_m(x, X_j)} = \frac{\sum_{X_j \in J} f(X_j)}{\sum_{X_j \in J} 1_J(X_j)} \tag{3.4}$$

for $x \in J$. Estimators of this general structure are called Nadaraya-Watson kernel estimators (Prakasa Rao [16]). Here we use the kernels generated by the Haar functions (see also Engel [7]).

**Theorem 3.3** *Consider a family of functions $\mathcal{F} \subseteq \mathcal{C}([0,1]^d)$ with range $[0,1]$ and capacity $h < \infty$ such that $\omega_\infty(f;r) = O(r^\alpha)$ as $r \to 0$, for $0 < \alpha \leq 1$. We assume that: (i) there exists a family of densities $\mathcal{P} \subseteq \mathcal{C}([0,1]^d)$; (ii) for each $p \in \mathcal{P}$, $\omega_\infty(p;r) = O(r^\alpha)$ as $r \to 0$, for $0 < \alpha \leq 1$; and (iii) there exists $\mu > 0$ such that for each $p \in \mathcal{P}$, $p(x) > \mu$. Suppose that the sample size, n, is larger than*

$$\max\left[2\left(\frac{2^m 9}{h!(\delta - \lambda)}\right)^{1/h}, \frac{4h^2}{\epsilon^4} 2^{4m+8}\right]$$

*where* $0 < \beta < d/2(\alpha + d)$, $m = \lceil \frac{\log n\beta}{d} \rceil$ *and* $\lambda = b\left(\frac{2}{\epsilon}\right)^{1/d+1/\alpha-1/2\alpha\beta} + b\left(\frac{4}{\epsilon(\mu-\epsilon)}\right)^{1/d+1/\alpha-1/2\alpha\beta}$. *Then for any* $f \in \mathcal{F}$, *we have* $P\left[\sup_x |f(x) - \tilde{f}_{m,n}(x)| > \epsilon\right] < \delta$.

**Proof:** We have $\tilde{f}_{m,n}(x) = \frac{\frac{1}{n}\sum_{X_j \in J} \frac{f(X_j)}{|J|}}{\frac{1}{n}\sum_{X_j \in J} \frac{1}{|J|}}$ where the denominator $\tilde{p}_{m,n}(x) = \frac{1}{n}\sum_{X_j \in J} \frac{1}{|J|}$ for $x \in J$ is the density estimator of [6]. Under the conditions of the theorem, Nadaraya [11] shows the following decomposition:

$$\begin{aligned} P\left[\sup_x |f(x) - \tilde{f}_{m,n}(x)| > \epsilon\right] \leq\ & P\left[\sup_x |f(x)p(x) - \tilde{f}_{m,n}(x)| > \frac{\epsilon(\mu-\epsilon)}{2}\right] \\ & + P\left[\sup_x |p(x) - \tilde{p}_{m,n}(x)| > \frac{\epsilon(\mu-\epsilon)}{2}\right] \\ & + P\left[\sup_x |p(x) - \tilde{p}_{m,n}(x)| > \epsilon\right]. \end{aligned}$$

We use Ciesielski's estimate [6] to bound the second and third terms by the application of the following:

$$\begin{aligned} P\left[\sup_x |p(x) - \tilde{p}_{m,n}(x)| > x\right] \leq\ & P\left[\sup_x |p(x) - P_m p(x)| > x/2\right] \\ & + P\left[\sup_x |P_m p(x) - \tilde{p}_{m,n}(x)| > x/2\right]. \end{aligned}$$

The first term is made zero by choosing $C/2^{\alpha m} \leq x/2$ and the second term is upperbounded by $b(2/x)^{1/d+1/\alpha-1/2\alpha\beta}$ for suitable constant $b$ (from proof of Theorem 3.13 of [6]). Now we have the first term of Nadaraya's decomposition bounded by

$$\begin{aligned} P\left[\sup_x |\tilde{f}_{m,n}(x) - f(x)p(x)| > y\right] \leq\ & P\left[\sup_x |P_m fp(x) - f(x)p(x)| > y/2\right] \\ & + P\left[\sup_x |\phi_n(x) - P_m fp(x)| > y/2\right]. \end{aligned}$$

The first term in the right hand side can be made zero by suitably choosing $m$, and the second term is estimated using the finite capacity:

$$P\left[|P_m fp(X) - \hat{f}_{m,n}| > y/2\right] \leq 2^m P\left[\sup_{J \in B_m} \left[|P_m fp(X) - \hat{f}_{m,n}(X)| 1_J(X)\right] > y/2^{m+1}\right].$$

Note that for $x \in J$, $P_m fp(x) = \frac{1}{|J|}\int_J f(x)p(x)dx$ which is the expectation of $f(x)1_J(x)$. Then $\hat{f}_{m,n}(x) = \frac{1}{n}\sum_{X_i \in J} f(X_j) = \frac{1}{n}\sum_{i=1}^{n} f(X_i)1_J(X_i)$ is the empirical mean of the the function $f(x)1_J(x)$. As in the case of Theorem 3.1, we apply Vapnik's bound to obtain

$$P\left[\sup_x |f(x) - \tilde{f}_{m,n}(x)| > \epsilon\right] \leq \lambda + \frac{2^{m+h}9}{h!} n^h e^{-n\epsilon^2/2^{2m+4}}$$

where $\lambda = b\left(\frac{2}{\epsilon}\right)^{1/d+1/\alpha-1/2\alpha\beta} + b\left(\frac{4}{\epsilon(\mu-\epsilon)}\right)^{1/d+1/\alpha-1/2\alpha\beta}$. The rest is as in Theorem 3.2. □

### 3.4 Computing the Estimates

Computation of $\hat{f}_{m,n}(x)$ or $\tilde{f}_{m,n}(x)$ at given $x$ involves obtaining the local sum of $f(X_i)$'s that are contained in $J$ containing $x$. The range-tree (see Preparata and Shamos [14]) can be constructed to store the cells $J$ that contain at least one $X_i$; with each such cell we store the number of the $X_i$'s that are contained in $J$ and the sum of the corresponding $f(X_i)$'s. This computation can be achieved by known methods [14], and the values of $J$ containing $x$ can be retrieved in $O((\log n)^d)$ time, and then $\hat{f}_{m,n}(x)$ or $\tilde{f}_{m,n}(x)$ can be computed in additional constant time. This same structure can be used to store the training sample of each $J$; once $J$ containing $x$ has been identified, $\bar{f}_{m,n}(x)$ can be computed in linear time. The following result directly follows from the results on range-tree [14].

**Theorem 3.4** *Based on a preprocessing in $O(n(\log n)^{d-1})$ time, resulting in a structure of size $O(n(\log n)^{d-1})$, the estimator $\hat{f}_{m,n}(x)$ or $\tilde{f}_{m.n}(x)$ for given $x$ can be computed in $O((\log n)^d)$ time. With no preprocessing, for given $x$, $\hat{f}_{m,n}(x)$, $\bar{f}_{m,n}(x)$, and $\tilde{f}_{m,n}(x)$ can be computed in $O(n)$ time.*

### 3.5 $\mathcal{L}^\infty$-Functions

Similar results can be shown for functions in $\mathcal{L}^\infty([0,1]^d)$. Recall that $\mathcal{C}([0,1]^d) \subset \mathcal{L}^\infty([0,1]^d)$ and the latter allows for discontinuities in the functions. Only minor changes to the proofs of last section are needed to establish the following result.

**Theorem 3.5** *Consider a family of functions $\mathcal{F} \subseteq \mathcal{L}^\infty(Q)$, with range in $[0,1]$ except on a set of measure zero and capacity $h < \infty$ such that for every $f \in \mathcal{F}$, we have $\omega_\infty(f;r) = O(r^\alpha)$ as $r \to 0$, for $0 < \alpha \leq 1$. For any $X$ chosen according to distribution $P_X$, we have for any $f \in \mathcal{F}$*

$$P\left[E|f(X) - \bar{f}_{m,n}(X)| > \epsilon\right] < \delta \qquad \text{and} \qquad P\left[|f(X) - \hat{f}_{m,n}(X)| > \epsilon\right] < \delta$$

*for the sample sizes given in Theorem 3.1 and 3.2 respectively. Under the conditions of Theorem 3.3 with $\mathcal{C}(Q)$ replaced by $\mathcal{L}^\infty(Q)$, for any $f \in \mathcal{F}$ and for sufficiently large $n$ like in Theorem 3.3, we have $P\left[\| f - \tilde{f}_{m,n} \|_\infty > \epsilon\right] < \delta$.*

## 4 Approximation to Neural Networks

Consider that $\mathcal{F}_W \subseteq \mathcal{F}$, i. e. the class $\mathcal{F}$ includes all functions approximated by feedforward networks of the form defined in Section 2. We now show that the nearest neighbor rule and regressogram provide approximations to the best possible neural network in the sense of Eq. (2.2). We show a technically stronger result that can be illustrated as follows for $\bar{f}_{m,n}$:

$$P[I(\bar{f}_{m,n}) - I(f) > \epsilon] < \delta$$

and $I(f_w^*) \geq I(f) = 0$, which implies $P[I(\bar{f}_{m,n} - I(f_w^*) > \epsilon] < \delta$. This result means that the nearest neighbor rule can PAC approximate the unknown function $f$ with parameters $\epsilon$ and $\delta$ even when the best neural network fails to (but, $\bar{f}_{m,n}$ can do no worse than $f_w^*$ in the PAC sense with parameters $\epsilon$ and $\delta$). Same reasoning applies to regressogram.

**Theorem 4.1** *Consider a family of continuous functions $\mathcal{F} \subseteq \mathcal{C}([0,1]^d)$, with range $[0,1]$ such that for every $f \in \mathcal{F}$, we have $\omega_\infty(f;r) \leq kr$ as $r \to 0$, for $0 < \alpha \leq 1$. Further let $k = \frac{\gamma abl}{4}$ where $a = \max_i |a_i|$ and $b = \max_{i,j} |b_{ij}|$. Then the nearest neighbor rule $\bar{f}_{m,n}$ approximates the best possible feedforward network $f_w^*$ such that*

$$P[I(\bar{f}_{m,n}) - I(f_w^*) > \epsilon] < \delta$$

*given a sample of size*

$$\frac{2^{2md}a^2b^2l^2d^3e}{\epsilon^2}\ln^2\left(\frac{2^{3md}a^2b^2l^2d^3e}{\delta\epsilon^2}\right)$$

*where $m = \frac{1}{\alpha}\log(4C/\epsilon\delta)$ is larger than a suitable $m_0$. Consider that $\mathcal{F}$ has capacity $h < \infty$ and $\omega_\infty(f;r) \leq kr^\alpha$, such that $k = \frac{\gamma abl}{4}$. The regressogram $\hat{f}_{m,n}$ satisfies*

$$P[I(\hat{f}_{m,n}) - I(f_w^*) > \epsilon] < \delta$$

*given a sample of size stated in Theorem 3.2.*

**Proof:** We first estimate an upper bound on the Lipschitz constant of $f_w$. Let us expand $f_w(x)$ as $\sum\limits_{j=1}^{l} a_j\sigma(\sum\limits_{i=1}^{d} b_{ji}x_i + t_j)$. The estimate on the Lipschitz constant can be obtained by maximizing the partial derivative $\frac{\partial f_w}{\partial x_j}$. First note that

$$\frac{\partial\sigma(z)}{\partial z} = \gamma\sigma(z)[1-\sigma(z)] \leq \gamma/4$$

since the right hand side is maximized at $\sigma(z) = 1/2$. Then

$$\frac{\partial f_w}{\partial x_j} = \sum_{i=1}^{l} a_i\sigma'(\sum_{i=1}^{d} b_{ji}x_i + t_j)b_{ij} \leq \frac{\gamma abl}{4}.$$

By the hypothesis, $\mathcal{F}_W \subseteq \mathcal{F}$ which implies $I(f_w^*) \geq I(f) = 0$. Then we have $P[I(\bar{f}_{m,n}) - I(f) > \epsilon] < \delta$ by Theorem 3.1, which implies the theorem for the nearest neighbor rule. The result for regressogram is similar by noting that the condition $|f(X) - \hat{f}(X)| < \epsilon$ implies $E|f(X) - \hat{f}(X)| < \epsilon$, since the domain is $[0,1]^d$ with measure 1. □

We now state the result based on the second cost functional which can be shown along the lines of Theorem 4.1.

**Theorem 4.2** *Under the hypothesis of Theorem 3.3, with the additional condition that $\omega_\infty(f;r) \leq kr^\alpha$ as $r \to 0$, for $0 < \alpha \leq 1$ and $k = \frac{\gamma abl}{4}$ where $a = \max\limits_i |a_i|$ and $b = \max\limits_{i,j} |b_{ij}|$ we have*

$$P[I_\infty(\bar{f}_{m,n}) - I_\infty(f_w^*) > \epsilon] < \delta$$

*given a sample of size specified in Theorem 3.3.* □

We note that boundedness of capacity of neural networks of this type can be deduced from the results of Macintyre and Sontag [10], and, in principle, can be used to obtain sample bounds along the lines of Theorem 4.1. However, the problem of computing $\hat{f}_w \in \mathcal{F}_W$ that PAC approximates optimal $f_w^* \in \mathcal{F}_W$ involves the loading problem which is NP-complete [19], whereas the three estimators above are linear-time computable. It is important, however, to note that the sample sizes of various cases for a fixed $(\epsilon, \delta)$ are not the same.

## 5 Conclusions

The PAC learning of smooth functions from finite samples by using feedforward neural networks has been "reduced" to the statistical estimation of functions via: (i) nearest neighbor, (ii) local averaging, and (iii) Nadaraya-Watson estimators. This reduction has to be understood in the sense that — for a sufficiently large sample — one can replace the neural networks estimators by any of the statistical estimators above, with no loss of performance. The three estimators are computed in the Haar function representation that lends itself easily to constructive proofs and convenient implementations. Our method relies on smoothness properties of the functions to yield estimators computable in linear time and yet guarantee PAC learning conditions. Although neither the statistical estimators nor the Haar system are new, non-trivial synthesis of techniques allowed us to obtain the following practically important results:

— under mild smoothness conditions on the function and/or the density we obtain stronger guarantees for the error than those based on capacity (or related combinatorial parameters) alone;

— smoothness properties are easier to infer from data than the capacity of the family;

— the results provide a deeper understanding of the approximation properties of neural networks, by relating them to well-known statistical estimators; and

— unlike the general PAC solutions that usually require solving NP-hard problems, the proposed estimators can be computed in linear time.

Generalizations of these results to PAC learning of functions by orthogonal systems is pursued elsewhere [15]. Tighter estimates and lower bounds for the sample sizes as well as a more precise relationship between smoothness guarantees and capacity for specific classes of functions are topics of future study.

## Acknowledgements

The authors gratefully acknowledge the guidance of Prof. Leo Brieman, Prof. Anselm Blumer and Dr. V. R. R. Uppuluri in valuable discussions that provided the foundation for this work. This research is sponsored by the Engineering Research Program of the Office of Basic Energy Sciences, of the U.S. Department of Energy, under Contract No. DE-AC05-84OR21400 with Lockheed Martin Energy Systems, Inc.

## References

[1] M. Anthony. Probabilistic analysis of learning in artificial neural networks: The PAC model and its variants. NeuroCOLT Technical Report Series NC-TR-94-3, Royal Holloway, University of London, 1994.

[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] 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.

[4] L. Brieman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees.* Wadsworth, Belmont, CA, 1984.

[5] B. Cheng and D. M. Titterington. Neural networks: A review from a statistical perspective. *Statistical Science*, 9(1):2–54, 1994.

[6] Z. Ciesielski. Haar system and nonparametric density estimation in several variables. *Probability and Mathematical Statistics*, 9:1–11, 1988.

[7] J. Engel. A simple wavelet approach to nonparametric regression from recursive partitioning schemes. *Journal of Multivariate Analysis*, 49:242–254, 1994.

[8] K. Fukunaga. *Introduction to Statistical Pattern Recognition.* Academic Press, 1990.

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

[10] A. Macintyre and E. D. Sontag. Finiteness results for sigmoidal neural networks. In *Proc. 25th Annual ACM Symp. on Theory of Computing*, pages 325–334. 1993.

[11] E. A. Nadaraya. Remarks on non-parametric estimates for density functions and regression curves. *Theory of Probability and Applications*, 15:134–137, 1970.

[12] V. Nedeljkovic. A novel multilayer neural networks training algorithm that minimizes the probability of classification error. *IEEE Transactions on Neural Networks*, 4(4):650–659, 1993.

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

[14] F. P. Preparata and M. I. Shamos. *Computational Geometry: An Introduction.* Springer-Verlag, 1985.

[15] H. Qiao, N. S. V. Rao, and V. Protopopescu. PAC learning of a class of functions by orthogonal series, 1995. Oak Ridge National Laboratory, manuscript under preparation.

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

[17] N. S. V. Rao. Fusion methods for multiple sensor systems with unknown error densities. *Journal of Franklin Institute*, 331B(5):509–530, 1995.

[18] N. S. V. Rao, V. Protopopescu, R. C. Mann, E. M. Oblow, and S. S. Iyengar. Learning algorithms for feedforward networks based on finite samples. Technical Report ORNL/TM-12819, Oak Ridge National Laboratory, September 1994. to appear in IEEE Trans. on Neural Networks.

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

[20] Y. Sakai, E. Takimoto, and A. Maruoka. Proper learning algorithm for functions of $k$ terms under smooth distributions. In *Proc. of 8th Ann. ACM Conf. on Computational Learning Theory*, 1995.

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

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

[23] V. N. Vapnik. Inductive principles of the search for empirical dependences. In *Proceedings of Second Ann. Workshop on Computational Learning Theory*, pages 3–21, 1989.

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

[25] H. White. Some asymptotic results for learning in single hidden layer feedforward network models. *Journal of American Statistical Association*, 84:1008–1013, 1989.

## DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.