On Overfitting, Generalization, and Randomly Expanded Training Sets

George N. Karystinos, *Student Member, IEEE,* and Dimitris A. Pados, *Member, IEEE*

***Abstract*—An algorithmic procedure is developed for the random expansion of a given training set to combat overfitting and improve the generalization ability of backpropagation trained multilayer perceptrons (MLPs). The training set is K-means clustered and locally most entropic colored Gaussian joint input–output probability density function (pdf) estimates are formed per cluster. The number of clusters is chosen such that the resulting overall colored Gaussian mixture exhibits minimum differential entropy upon global cross-validated shaping. Numerical studies on real data and synthetic data examples drawn from the literature illustrate and support these theoretical developments.**

***Index Terms*—Backpropagation, clustering methods, entropy, Gaussian distributions, multilayer perceptrons (MLPs), stochastic approximation, stochastic processes.**

I. Introduction

MULTILAYER perceptron (MLP) networks have been used extensively in the past for the functional approximation of continuous nonlinear mappings. For a given network structure the MLP can be viewed as a nonlinear parametrically described function. Successful functional approximation depends on the appropriate selection of the parameter values. This selection is usually made through supervised learning where a training set of input–output pairs is available and the network is trained to match this set according to some prespecified criterion function. When the criterion function is the sum of squared errors, the corresponding algorithm is the well-known backpropagation (BP) training procedure [1], [2].

The operational performance measure for the trained network is the error on future data outside the training set, also known as generalization error. This error may be undesirably large when, for example, the available training set size is too small in comparison with the network parameter set size. Practice has shown that direct minimization of the training error for a given fixed training set obtained by BP-type learning algorithms does not necessarily imply a corresponding minimization of the generalization error. Even worse, in most reported cases the decrease of the generalization error exhibited during the first few successive passes through the same set of examples (usually called epochs) may be followed by a steady increase. In the neural network literature this phenomenon is usually referred to as "overfitting."

Recently, it was noted [3], [4] that the algorithms that use instantaneous estimates of the error function to adapt system parameters (such as BP [1], [2] or Rosenblatt's perceptron rule [5]) are in essence direct applications of stochastic approximation procedures [6]–[8]. The theory of stochastic approximation indicates that these methods provide strongly consistent optimization [9]. In other words, given an infinite sequence of input–output pairs drawn from some joint probability distribution function (pdf), the induced sequence of network designs converges with probability one to a locally optimum design (in general global optimization cannot be guaranteed for nonlinear systems). This optimality is strictly with respect to the training input–output vectors' pdf. If the training sequence is drawn from the true input–output pdf, then local optimality is achieved in the minimum generalization error sense with probability one. Of course, in real-life applications the network designer is provided with only a finite set of training examples. Then, "data recycling" leads to the assignment of excessive probability mass on the exact points of the training set. As a result, the neural network "concentrates" more and more on these excessively weighted examples at the expense of poor generalization ability (overfitting).

The estimation of the generalization performance of MLP's is a problem of fundamental significance with great implications in the theory and applications of neural networks. A somewhat standard statistical technique for coping with the generalization error is *cross-validation*. The available training set is divided into subsets: the *training set* and the *validation* or *test set*. The data in the training set are used during the learning stage for the network adaptation, while the data in the test set are reserved for performance evaluation upon training (exactly as with group method of data handling (GMDH)-type algorithms [10]). To achieve statistically significant results, several independent data splits must be performed followed by lengthy training times. The single-split alternative with test-error monitoring comes with a significant waste of data (data in the test set are never used in the training procedure).

Given these limitations of the cross-validation method, there have been several attempts (for example, see [11]) to improve the generalization ability of MLPs by an automatic architecture selection process through the use of Akaike's information criterion [12], or Rissannen–Schwarz's minimum description length [13], [14], or Barron's predicted square error (PSE) [15] criterion. All these criteria lead to expressions that are linear in the number of free network parameters and are motivated by the theory of linear systems (for example, autoregressive or moving average models). Their optimality is based on the assumed linearity of the underlying system and the Gaussianity of the error signals. Unfortunately, both assumptions are violated in the context of neural networks. Moreover, the number of free system parameters does not determine uniquely the neural network architecture and even if we choose to accept the Gaussianity of the pertinent

Manuscript received November 13, 1998; revised April 4, 2000.

The authors are with the Department of Electrical Engineering, State University of New York, Buffalo, NY 14260-2050 USA (e-mail: cary@eng.buffalo.edu; pados@eng.buffalo.edu).

Publisher Item Identifier S 1045-9227(00)06028-8.

1045–9227/00$10.00 © 2000 IEEE

Authorized licensed use limited to: Zhejiang University. Downloaded on August 21,2020 at 08:33:14 UTC from IEEE Xplore. Restrictions apply.

signals, the mean square error (MSE) surface does not have in general a unique minimum as in the linear system models.

In view of these difficulties, researchers have followed a variety of different approaches known as *pruning*,[1] *weight sharing*,[2] and *complexity regularization*.[3] In this paper we look at the same problem but from a different—yet synergistic—point of view. We proceed with a random generalization/expansion of the available finite training set. Upon choosing appropriately the distribution characteristics of the random expansion process, an infinite sequence of artificial training input–output vectors is created which—when used for network training—is expected to combat overfitting. A related concept—as we will see in the sequel—was originally suggested in [18] and [19] in the form of adding "noise" to the training set. In particular, additive white Gaussian noise to each training input–output vector was considered in [20], based on the Parzen-Rosenblatt estimate [21]–[23] of the true input–output vector density. Instead, in this work we propose a locally most entropic estimate of the true *joint input–output* pdf. Expansion of the training set can then be achieved by drawing new random training vectors from this density estimate. The new training set is seen to avoid overfitting and improve the generalization ability of MLP networks when BP learning is applied. We conclude with the comment that the random expansion of the training set pursued in this work treats each input–output pair in the training set as an integral vector and accounts for the joint input–output distribution that governs its formation. This is in contrast to the work in [18]–[20] where independent noise is added to the input and output components and in sharp contrast to Tikhonov regularization procedures that were shown to be equivalent to training with additive noise to the input only [24], [25]. It is also satisfying to observe that the additive-white-noise estimate [20] or the colored-Gaussian density estimate [26] can be viewed as extreme special cases of the proposed locally most entropic estimation approach.

The rest of this paper is organized as follows. Section II is devoted to probabilistic, stochastic-approximation-based analysis of training set expansion methods for generalization improvement. In Section III we propose an information theoretic approach to the problem of random expansion of the training set. Supporting simulation results for functional approximation problems are presented in Section IV. Finally, some conclusions are drawn in Section V.

II. Training Versus Generalization Error

Let $\{\mathbf{Z}_n\}_{n=1}^{\infty}$ be a sequence of identically distributed and statistically independent random vectors where each $\mathbf{Z}_n = (\mathbf{X}_n, \mathbf{Y}_n) \in \mathcal{R}^{m+l}$, $\mathbf{X}_n \in \mathcal{R}^m$ and $\mathbf{Y}_n = q(\mathbf{X}_n) \in \mathcal{R}^l$, is viewed as an input–output pair measurement from an unknown continuous nonlinear mapping $q : \mathcal{R}^m \to \mathcal{R}^l$.

Let $g(\cdot;\mathbf{w}) : \mathcal{R}^m \to \mathcal{R}^l$ be a continuous mapping parametrically described in $\mathbf{w} \in \mathcal{R}^d$. The objective is to use $g(\cdot;\mathbf{w})$ to approximate $q(\cdot)$ in the MSE sense. More precisely, we wish to utilize the available sequence of data $\{\mathbf{Z}_n\}_{n=1}^{\infty}$ to identify a parameter vector $\mathbf{w}^o \in \mathcal{R}^d$ such that the MSE $E_{f_{\mathbf{X},\mathbf{Y}}}\{||g(\mathbf{X};\mathbf{w}^o) - \mathbf{Y}||^2\}$ is minimized where $E_{f_{\mathbf{X},\mathbf{Y}}}\{\cdot\}$ denotes statistical expectation with respect to the joint pdf $f_{\mathbf{X},\mathbf{Y}}$ of the random variables $\mathbf{X}$ (input) and $\mathbf{Y}$ (output). In other words

$$\mathbf{w}^o = \arg\min_{\mathbf{w}} E_{f_{\mathbf{X},\mathbf{Y}}}\{||g(\mathbf{X};\mathbf{w}) - \mathbf{Y}||^2\}. \tag{1}$$

Let $\phi[(\mathbf{X},\mathbf{Y}),\mathbf{w}] \triangleq \nabla_{\mathbf{w}}(||g(\mathbf{X};\mathbf{w}) - \mathbf{Y}||^2)$ and $M(\mathbf{w}) \triangleq E_{f_{\mathbf{X},\mathbf{Y}}}\{\phi[(\mathbf{X},\mathbf{Y}),\mathbf{w}]\}$. In general, $M(\mathbf{w}) = \mathbf{0}$ has multiple roots for an arbitrary nonlinear function g. If θ is a root, then under regularity conditions $M(\theta) = \mathbf{0}$ implies $\nabla_{\mathbf{w}} E_{f_{\mathbf{X},\mathbf{Y}}}\{||g(\mathbf{X};\theta) - \mathbf{Y}||^2\} = \mathbf{0}$ and, therefore, θ is an extremum of the examined MSE surface.

If $\{\alpha_n\}_{n=1}^{\infty}$ is a sequence of positive monotone decreasing scalars such that $\sum_{n=1}^{\infty} \alpha_n^2 < \infty$ and $\sum_{n=1}^{\infty} \alpha_n = \infty$, and $\mathbf{w}_1 \in \mathcal{R}^d$ is an arbitrary initial vector, then

$$\mathbf{w}_{n+1} = \mathbf{w}_n - \alpha_n \phi(\mathbf{Z}_n, \mathbf{w}_n), \quad n = 1, 2, \ldots \tag{2}$$

defines a (nonstationary) Markov chain $\{\mathbf{w}_n\}_{n=1}^{\infty}$. In [6] it was shown that under certain conditions, $\mathbf{w}_n$ converges in the mean square sense to a root of the equation $M(\mathbf{w}) = \mathbf{0}$ and minimizes the MSE with respect to the underlying distribution $f_{\mathbf{X},\mathbf{Y}} = f_{\mathbf{Z}}$ *of the given infinite sequence* $\{\mathbf{Z}_n\}_{n=1}^{\infty}$. Convergence of $\mathbf{w}_n$ to a minimum (or oscillation between multiple minima) with probability one was shown, under more general conditions, in [9].

Let us now consider an MLP network with input dimension m and output dimension l and let $g(\cdot;\mathbf{w})$, $\mathbf{w} \in \mathcal{R}^d$, represent the input–output operation of the network, where $\mathbf{w}$ is the parameter vector that includes all network weights and thresholds. Given a data sequence $\{\mathbf{Z}_n\}_{n=1}^{\infty}$, we can train the network through (2) to minimize the MSE $E_{f_{\mathbf{X},\mathbf{Y}}}\{||g(\mathbf{X};\mathbf{w}) - \mathbf{Y}||^2\}$. The resulting procedure is theoretically equivalent to the familiar BP algorithm.[4] Since MLP's are nonlinear structures, recursion (2) will in general lead the network to a local minimum of the MSE surface.

Of course, in reality only a finite data set $\{\mathbf{Z}_n\}_{n=1}^{N}$ is available and limited availability of data is commonly treated by data recycling. Data recycling means that the data set $\{\mathbf{Z}_n\}_{n=1}^{N}$ is repeatedly fed to the network through recursion (2), either in the original order or after shuffling. We recall that in the neural-network literature each pass over the data set is known as an epoch. It is important to observe that infinite data recycling, that is training with infinitely many epochs, corresponds to system optimization with data drawn from the density function

$$f'_{\mathbf{Z}}(\mathbf{z}) = \frac{1}{N} \sum_{n=1}^{N} \delta(\mathbf{z} - \mathbf{Z}_n) \tag{3}$$

where $\delta(\cdot)$ denotes the delta generalized function. This observation offers a direct statistical interpretation of overfitting phe-

[1]Pruning is the process of eliminating connections from a fully connected and trained MLP. After this, the reduced size network is retrained. A survey of pruning algorithms can be found in [16].

[2]The idea behind weight sharing [17] is to arrange the hidden layer nodes in groups with common weight values, where each group processes only a local region of the input.

[3]Complexity regularization methods add a second term to the usual sum-of-square-error criterion function. This additional term may penalize the existence of a large number of weights, or of large-valued weights, or both.

[4]The BP algorithm evaluates efficiently the gradient of the norm-square-error $\phi(\cdot,\cdot)$ through an ordered sequence of partial derivatives that accounts for the specific MLP architecture.

Authorized licensed use limited to: Zhejiang University. Downloaded on August 21,2020 at 08:33:14 UTC from IEEE Xplore. Restrictions apply.

nomena. Indeed, direct substitution of (3) in (1) shows that network training through (2) leads to convergence of $\mathbf{w}_n$ to $\mathbf{w}^{o\prime} = \arg\min_{\mathbf{w}} \sum_{n=1}^{N} \|g(\mathbf{X}_n;\mathbf{w}) - \mathbf{Y}_n\|^2$, where $(\mathbf{X}_n, \mathbf{Y}_n) = \mathbf{Z}_n$, for $n = 1, 2, \ldots, N$. Of course, $\sum_{n=1}^{N} \|g(\mathbf{X}_n;\mathbf{w}) - \mathbf{Y}_n\|^2$ is what we call the "training error" of the network to the given training set $\{\mathbf{Z}_n\}_{n=1}^{N}$ and we conclude that data recycling of this form leads to the minimization of the training error. However, no attempt is made whatsoever to project in any reasonable way data characteristics outside the given set.

Theoretically, to minimize the generalization error $E_{f_{\mathbf{X},\mathbf{Y}}}\{\|g(\mathbf{X};\mathbf{w}) - \mathbf{Y}\|^2\}$ of the network we need an infinite data sequence $\{\mathbf{Z}_n\}_{n=1}^{\infty}$ drawn from the true input–output distribution $f_{\mathbf{X},\mathbf{Y}}$ of the unknown mapping q. In the sequel, we focus on finite training sets and based on a given finite set we wish to develop alternatives to the $f'_{\mathbf{Z}}(\mathbf{z})$ density function in (3), say $f^{(1)}_{\mathbf{X},\mathbf{Y}}$, such that training with data drawn from $f^{(1)}_{\mathbf{X},\mathbf{Y}}$ may prevent overfitting and lead to networks with superior generalization ability. Along these lines, it was shown in [20] that

$$\sup_{\mathbf{w}} \left| E_{f^{(1)}_{\mathbf{X},\mathbf{Y}}}\{\|g(\mathbf{X};\mathbf{w}) - \mathbf{Y}\|^2\} - E_{f_{\mathbf{X},\mathbf{Y}}}\{\|g(\mathbf{X};\mathbf{w}) - \mathbf{Y}\|^2\} \right| \le C_\epsilon \cdot (\|f^{(1)}_{\mathbf{X},\mathbf{Y}} - f_{\mathbf{X},\mathbf{Y}}\|_1)^{\frac{\epsilon}{2+\epsilon}} \tag{4}$$

where $\|\cdot\|_1$ denotes the L_1 norm with respect to the Lebesgue measure, ϵ is a positive constant, and $C_\epsilon > 0$ is a scalar that depends on ϵ. Given that [27]

$$D(f_1\|f_2) \ge \frac{1}{2\ln 2}\|f_1 - f_2\|_1^2 \tag{5}$$

where $D(f_1 \,\|\, f_2)$ denotes the relative entropy [also known as the Kullback–Leibler (K–L) distance] between two arbitrary pdfs f_1 and f_2, we obtain

$$\sup_{\mathbf{w}} \left| E_{f^{(1)}_{\mathbf{X},\mathbf{Y}}}\{\|g(\mathbf{X},\mathbf{w}) - \mathbf{Y}\|^2\} - E_{f_{\mathbf{X},\mathbf{Y}}}\{\|g(\mathbf{X},\mathbf{w}) - \mathbf{Y}\|^2\} \right| \le K_\epsilon \cdot D\left(f_{\mathbf{X},\mathbf{Y}} \,\middle\|\, f^{(1)}_{\mathbf{X},\mathbf{Y}}\right)^{\frac{\epsilon}{2(2+\epsilon)}} \tag{6}$$

where $K_\epsilon = C_\epsilon \cdot (2\ln 2)^{\epsilon/2(2+\epsilon)}$ is a positive scalar that depends on ϵ. This inequality shows that $|E_{f^{(1)}_{\mathbf{X},\mathbf{Y}}}\{\|g(\mathbf{X};\mathbf{w}) - \mathbf{Y}\|^2\} - E_{f_{\mathbf{X},\mathbf{Y}}}\{\|g(\mathbf{X};\mathbf{w}) - \mathbf{Y}\|^2\}|$ can be made small by designing a density function $f^{(1)}_{\mathbf{X},\mathbf{Y}}$ such that the K–L distance between $f^{(1)}_{\mathbf{X},\mathbf{Y}}$ and the true density $f_{\mathbf{X},\mathbf{Y}}$ is minimized.

In [20] it was argued that the generalization ability of a BP-trained neural network can be improved by introducing additive noise to the training samples. In the framework of our presentation this corresponds to a network trained with samples drawn from the pdf

$$f_{N,\sigma}(\mathbf{z}) = \frac{1}{N}\sum_{n=1}^{N} \frac{1}{\sigma^{m+l}} K\left(\frac{\mathbf{z} - \mathbf{Z}_n}{\sigma}\right) \tag{7}$$

where $K(\cdot)$ is a pdf "kernel" and $\sigma > 0$. In particular, since additive white Gaussian noise was considered in [20], the kernel was effectively set to be a Gaussian density with zero mean and covariance matrix equal to the identity matrix. We note that (7) is also known as the Parzen–Rosenblatt estimate [21], [22] of the true training vector density $f_{\mathbf{X},\mathbf{Y}}$ and in [23] it was shown that, for each $\mathbf{z} \in \mathcal{R}^{m+l}$, $f_{N,\sigma}(\mathbf{z}) \to f_{\mathbf{X},\mathbf{Y}}(\mathbf{z})$ in probability as $N \to \infty$ under some general conditions.

Another approach was considered in [26] in a different context, where a colored Gaussian pdf with a sample average mean vector and a sample average covariance matrix estimate was used for the approximation of the true distribution of a given finite population. In the following section we develop a new methodology for the random expansion of a given training set that encompasses these two previous pdf estimation procedures as special extreme cases.

III. Locally Most Entropic Expansion of the Training Set

We begin this section with two propositions that set the stage for later developments. The first proposition comes directly from information theory [27].

Proposition 1: The pdf that maximizes the differential (Shannon) entropy $h(f) \triangleq -\int_{\mathcal{R}^{m+l}} f(\mathbf{z}) \log f(\mathbf{z})\, d\mathbf{z}$ over all pdfs f with support set $\mathcal{R}^{m+l}$, mean vector $\mathbf{U} \in \mathcal{R}^{m+l}$, and covariance matrix $R_{(m+l)\times(m+l)}$ is the $(m+l)$-dimensional normal density with these moments, denoted by $f_{\mathrm{ME}} = \mathcal{N}(\mathbf{U}, R)$ where

$$f_{\mathrm{ME}}(\mathbf{z}) = \frac{1}{(\sqrt{2\pi})^{m+l}|R|^{\frac{1}{2}}} e^{-\frac{1}{2}(\mathbf{z}-\mathbf{U})^T R^{-1}(\mathbf{z}-\mathbf{U})} \tag{8}$$

for any $\mathbf{z} \in \mathcal{R}^{m+l}$ (the superscript ^{T} denotes the vector transpose operation and $|R|$ denotes the determinant of R). □

To meet the special needs of the problem that we investigate in this work, we find it necessary to attempt a generalization of the well-known result of Proposition 1. Along these lines, let us assume that a data vector $\mathbf{Z}$ may come from either one of K "hypotheses" $H_1, H_2, \ldots, H_K$, with corresponding prior probabilities $\pi_1, \pi_2, \ldots, \pi_K$. Let us also assume that each hypothesis $H_k, k = 1, 2, \ldots, K$, represents local input–output data characteristics in a subset of the overall input–output space $\mathcal{R}^{m+l}$ that occurs with probability $\pi_k, k = 1, 2, \ldots, K$. The following proposition lays the foundation for the development of a systematic procedure for the random expansion of a given training set.

Proposition 2: We consider a random data vector $\mathbf{Z}$ that, with probability π_k, comes from hypothesis H_k with support set $\mathcal{R}^{m+l}$, mean $\mathbf{U}_k$, and covariance matrix $R_k, k = 1, 2, \ldots, K$. Enforcing the constraint of maximum differential entropy under every hypothesis $H_k, k = 1, 2, \ldots, K$, the unconditional pdf of $\mathbf{Z}$ is the Gaussian mixture

$$f_{\mathrm{LME}} = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{U}_k, R_k) \tag{9}$$

that we call the locally (or conditionally) most entropic (LME) density. □

Consider now a finite set $\{\mathbf{Z}_n\}_{n=1}^{N}$ of samples $\mathbf{Z}_n = (\mathbf{X}_n, \mathbf{Y}_n) \in \mathcal{R}^{m+l}$ that is given to us for the training of an MLP network. For a given K, we proceed with

Authorized licensed use limited to: Zhejiang University. Downloaded on August 21,2020 at 08:33:14 UTC from IEEE Xplore. Restrictions apply.

the partitioning of the available training set into K clusters $C_k, k = 1, 2, \ldots, K$, using some minimum distortion clustering procedure [35]. To establish a correspondence in terminology with Proposition 2, let hypothesis H_k refer to data membership in cluster $C_k, k = 1, 2, \ldots, K$. Let N_k be the number of data vectors $\mathbf{Z} = (\mathbf{X}, \mathbf{Y})$ in cluster $C_k, k = 1, 2, \ldots, K$. Unbiased estimates for the parameters $\mathbf{U}_k$, R_k, and $\pi_k, k = 1, 2, \ldots, K$, in (9) can be readily obtained in the form of sample averages [28] as seen in (10)–(12), shown at the bottom of the page. We note that for the single-vector cluster case ($N_k = 1$), no unbiased estimate for R_k can be defined and $\hat{R}_k$ is forced to be equal to the $(m+l) \times (m+l)$ zero matrix $\mathbf{0}_{(m+l)\times(m+l)}$.

Under the optimality criterion of maximum entropy per local cluster, the core concept of this present work is the proposal for MLP training with data drawn from $\hat{f}_{\mathrm{LME}}$, defined as an estimate of (9) via (10)–(12). Certainly, several aspects of this proposed process require further examination. Namely, 1) the possibility of facing noninvertible estimates $\hat{R}_k$; 2) an optimality criterion for the automatic selection of the number of clusters K; and 3) the design of a simple and effective implementation procedure that relies only on trivial random number generators, are all issues of great importance.

We begin with a suggested generalization of the covariance matrix estimator in (13), shown at the bottom of the page, where I is the identity matrix. We note that when $\sigma_M^2 = 1$ and $\sigma_{DL,k}^2 = 0$ for all k, (13) degenerates to (11) and when $\sigma_{DL,k}^2 = 0$ at least $m + l$ data samples need to be in C_k for the cluster covariance matrix estimate $\hat{R}_k$ to be invertible with probability one [29]. Otherwise, some strictly positive value $\sigma_{DL,k}^2$ is necessary to achieve invertibility. In engineering practice this biasing of the covariance matrix estimator has been known as diagonal loading (DL). In the context of our presentation diagonal loading (that is a choice of $\sigma_{DL,k}^2 > 0$) is equivalent to white Gaussian noise expansion of the cluster data about their mean $\hat{\mathbf{U}}_k$, with variance $\sigma_{DL,k}^2$.

The other parameter σ_M^2 in (13) offers the opportunity for biasing the "peakedness" of the Gaussian mixture distribution in (9) when $\sigma_M^2 \neq 1$ is chosen. For the selection of σ_M we propose a method based on equal likelihood cross-validation at the cluster level[5] (maximum likelihood cross-validation [31]–[33] has been used extensively in the past, for example for the selection of the variance parameter in (7) for the simple white Gaussian kernel case [20]). We assume first that $K > 1$. Then, for each cluster $C_k, k = 1, 2, \ldots, K$, we form the pdf

$$f_{\mathbf{Z}/\mathbf{Z}\notin C_k} = \frac{1}{N - N_k} \sum_{\substack{j=1 \\ j\neq k}} N_j \cdot \mathcal{N}(\hat{\mathbf{U}}_j, \hat{R}_j), \quad k = 1, 2, \ldots, K \tag{14}$$

as the mixture of the "other-cluster" estimates. We denote the overall σ_M-parameterized LME density by

$$f_{\sigma_M} = \frac{1}{N} \sum_{j=1}^{K} N_j \cdot \mathcal{N}\left(\hat{\mathbf{U}}_j, \sigma_M^2 \hat{R}_j\right), \quad k = 1, 2, \ldots, K \tag{15}$$

and we let C_{K_n} denote the cluster that $\mathbf{Z}_n$ belongs to, $n = 1, 2, \ldots, N$. Under the assumption of statistically independent training vectors, we calculate the following joint densities:

$$\begin{aligned} f^*(\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_N) &\triangleq \\ & f_{\mathbf{Z}_1,\mathbf{Z}_2,\ldots,\mathbf{Z}_N/\mathbf{Z}_1\notin C_{K_1},\mathbf{Z}_2\notin C_{K_2},\ldots,\mathbf{Z}_N\notin C_{K_N}}(\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_N) \\ &= \prod_{n=1}^{N} f_{\mathbf{Z}/\mathbf{Z}\notin C_{K_n}}(\mathbf{z}_n) \end{aligned} \tag{16}$$

and

$$f_{\sigma_M}^*(\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_N) \triangleq \prod_{n=1}^{N} f_{\sigma_M}(\mathbf{z}_n). \tag{17}$$

Finally, we choose the $\sigma_M > 0$ value that is the minimum root of the equation

$$f^*(\mathbf{Z}_1, \mathbf{Z}_2, \ldots, \mathbf{Z}_N) = f_{\sigma_M}^*(\mathbf{Z}_1, \mathbf{Z}_2, \ldots, \mathbf{Z}_N). \tag{18}$$

Since no cross-validated pdf [according to the setup in (14)] can be defined for the case $K = 1$, the value $\sigma_M = 1$ is forced for this case.

For the development of an algorithmic procedure that selects an "appropriate" value for the number of clusters K, we propose a minimum entropy choice (criterion) among all LME pdf solutions for each $K \in \{1, 2, \ldots, N\}$. This criterion is described as follows.

Criterion 1 (Min Entropy LME Rule): For a given training data set of size N, the minimum-entropy optimum value of the number of clusters $K \in \{1, 2, \ldots, N\}$ is defined as the value that induces an LME density function f_{LME} with min-

[5]Another worth investigating procedure for maximum-likelihood selection of the shaping parameter σ_M may be expectation-maximization (EM) [30]: $\sigma_M = \arg\max_{\sigma_M} E\{\log f_{\mathrm{LME}}(\mathbf{z}/\sigma_M)\}$ where the expectation step draws data from the original training set.

$$\hat{\mathbf{U}}_k = \frac{1}{N_k} \sum_{\mathbf{Z}_n \in C_k} \mathbf{Z}_n, \quad k = 1, 2, \ldots, K \tag{10}$$

$$\hat{R}_k = \begin{cases} \frac{1}{N_k - 1} \sum_{\mathbf{Z}_n \in C_k} (\mathbf{Z}_n - \hat{\mathbf{U}}_k)(\mathbf{Z}_n - \hat{\mathbf{U}}_k)^T, & N_k > 1 \\ \mathbf{0}_{(m+l)\times(m+l)}, & N_k = 1 \end{cases}, \quad k = 1, 2, \ldots, K \tag{11}$$

$$\hat{\pi}_k = \frac{N_k}{N}, \quad k = 1, 2, \ldots, K. \tag{12}$$

$$\hat{R}_k(\sigma_M, \sigma_{DL,k}) = \begin{cases} \sigma_M^2 \left[\frac{1}{N_k - 1} \sum_{\mathbf{Z}_n \in C_k} (\mathbf{Z}_n - \hat{\mathbf{U}}_k)(\mathbf{Z}_n - \hat{\mathbf{U}}_k)^T + \sigma_{DL,k}^2 I \right], & N_k > 1 \\ \sigma_M^2 \sigma_{DL,k}^2 I, & N_k = 1 \end{cases}. \tag{13}$$

Authorized licensed use limited to: Zhejiang University. Downloaded on August 21,2020 at 08:33:14 UTC from IEEE Xplore. Restrictions apply.

imum differential entropy $h(f_{\mathrm{LME}})$. We recall that for every $K \in \{1, 2, \ldots, N\}$ the LME density function is defined by (9) through (10), (12), and (13)–(18). □

Several clustering procedures exist for the partitioning of a set into K clusters. Arguably, the K-means algorithm [36] is among the most widely used and for infinite data sets is known to converge with probability one to a local minimum of the *sum-of-squared-errors* distortion function. For finite sets of size N, to obtain a distortion value close to the globally minimum several runs of the batch K-means procedure may be necessary, each one starting from a different initial set of clusters out of the $\mathcal{S}_N^{(K)} = (1/K!)\sum_{k=0}^{K}(-1)^{K-k}\binom{K}{k}k^N$ possible initial set combinations [37].

For a given K-cluster partition of the training set, computation of the underlying differential entropy $h(f_{\mathrm{LME}}) = -\int_{\mathcal{R}^{m+l}} f_{\mathrm{LME}}(\mathbf{z}) \log f_{\mathrm{LME}}(\mathbf{z})\, d\mathbf{z} = -E_{f_{\mathrm{LME}}}\{\log f_{\mathrm{LME}}(\mathbf{z})\}$ requires $(m + l)$-dimensional integration which is not always feasible. In such cases, Monte-Carlo sample-average calculation through J data points drawn from f_{LME}, $\hat{h}(f_{\mathrm{LME}}) = -(1/J)\sum_{j=1}^{J} \log f_{\mathrm{LME}}(\mathbf{Z}_j)$, is a practical alternative.

To summarize briefly the developments so far in this section, given a data set $\{\mathbf{Z}_n\}_{n=1}^{N}$ for the training of an MLP we consider the following general procedure: First, upon K-means clustering for a given number of clusters $K \in \{1, 2, \ldots, N\}$, we estimate from the data the LME pdf $f_{\mathrm{LME}}(K)$ given by (9), (10), (12), and (13)–(18). Then, we optimize with respect to the number of clusters K according to Criterion 1 and we identify the minimum entropy LME pdf $f_{\text{ME-LME}}$. Finally, we proceed with conventional MLP training (BP for example) with random training data drawn from $f_{\text{ME-LME}}$. It is satisfying to observe that for implementation purposes no special purpose random vector generator is needed. To draw data $\mathbf{Z}^{(i)}$, $i = 1, 2, \ldots$, from cluster $k \in \{1, 2, \ldots, K\}$ with pdf $\mathcal{N}(\hat{\mathbf{U}}_k, \hat{R}_k)$ where $\hat{\mathbf{U}}_k$ and $\hat{R}_k$ are given by (10) and (13), respectively, we write

$$\mathbf{Z}^{(i)} = \hat{\mathbf{U}}_k + \hat{L}_k \mathbf{s}^{(i)} \tag{19}$$

where $\mathbf{s}^{(i)}$ is an independently identically distributed (i.i.d.) vector sequence drawn from $\mathcal{N}(\mathbf{0}, I)$ and $\hat{L}_k$ is the Cholesky lower triangular matrix from the decomposition of the—possibly diagonally loaded by $\sigma_{DL,k}^2$ and preshaped by σ_M^2—covariance matrix estimate $\hat{R}_k$ for cluster C_k in (13). In other words, $\hat{L}_k$ is such that $\hat{R}_k = \hat{L}_k \hat{L}_k^T$. Of course, during MLP training data utilization per cluster maintains the relative frequency ratio $N_k/N = \hat{\pi}_k$, $k = 1, 2, \ldots, K$, in (12). As discussed in Section II, stochastic approximation arguments [3] show that if we train an MLP by such an infinite, randomly expanded training set we achieve strongly consistent MSE minimization under the $f_{\text{ME-LME}}$ joint input–output pdf.

We conclude this section with the comment that this newly developed random training set expansion procedure covers the approaches in [20] and [26] as special and extreme cases. Indeed, in our setup the work in [20] is the $K = N$ case, while the work in [26] is the $K = 1$ case. The following section is devoted to overfitting and generalization performance comparisons.

IV. Numerical and Simulation Studies

To illustrate the theoretical developments in the previous sections, we examine three different examples that involve the estimation of an unknown nonlinear mapping by a given MLP. Two synthetic data examples are taken directly from the pertinent literature [20], [34] and a real data case study is drawn from stock market financial records. In all cases the objective is to study the generalization behavior of a given MLP under 1) standard multiple-epoch BP training; 2) BP training with additive white Gaussian noise (AWGN) as in [20]; and 3) BP training with LME training set expansion as proposed in this work.

A. Single-Input/Single-Output Example

We revisit the sinusoidal function example in [20]

$$f(x) = 0.4 \sin x + 0.5, \quad x \in \mathcal{R}. \tag{20}$$

As in [20], we generate $N = 36$ independent input–output vectors $\mathbf{Z}_n = (X_n, Y_n), n = 1, 2, \ldots, 36$, with X_n uniformly distributed on $[-\pi, \pi]$ and $Y_n = f(X_n) + r_n$, where $r_n \sim \mathcal{N}(0, \sigma_{\mathrm{obs}}^2)$ accounts for "observation noise" with variance $\sigma_{\mathrm{obs}}^2 = 10^{-2}$. The resulting training set instance can be seen in Fig. 1. The network used in [20] for this estimation problem was a 1-13-1 feedforward MLP network with activation function $1/1 + e^{-t}$. The same MLP is considered here and all network parameters are randomly initialized uniformly on $[-1/2, 1/2]$. First, we perform standard BP training using the same 36-point training set of Fig. 1 during all epochs. Next, we apply the AWGN method of [20] where the original training set is used only during the first epoch, $i = 1$, and then, for future epochs, $i > 1$, 36 new random training data are generated per epoch according to $\mathbf{Z}_n^{(i)} = \mathbf{Z}_n + \mathbf{u}^{(i)}, n = 1, 2, \ldots, 36$, where $\mathbf{u}^{(i)} \sim \mathcal{N}(\mathbf{0}, \sigma^2 I)$, I is the 2×2 identity matrix and $\sigma^2 = 0.0262$ is chosen as in [20] (maximum likelihood cross-validation). Finally, we implement the proposed LME training set expansion procedure, that for the data set of Fig. 1 results to $K = 11$ clusters (identified in Fig. 1) and $\sigma_M^2 = 0.03$. Indeed, Fig. 2 shows the differential entropy of the LME density for the given data set for $1 \leq K \leq 21$. According to Criterion 1, $K = 11$ appears to be the most appropriate selection. The original training set $\mathbf{Z}_n, n = 1, \ldots, 36$ is used during the first epoch, $i = 1$, and for each future epoch, $i > 1$, 36 new random training data $\mathbf{Z}_n^{(i)}, n = 1, \ldots, 36$ are generated according to (19) and the cluster membership of the corresponding original data point $\mathbf{Z}_n$. In Fig. 3 we plot the target sinusoid function in (20) together with the original training set points and the MLP estimate derived after 100 000 epochs by standard BP training, AWGN training as in [20], and LME training as proposed herein.

Fig. 4 shows the induced generalization error as a function of the training epochs which is calculated *analytically* for all three methods. The error on the original training set of all procedures is also included as a reference. For increased credibility, the results in Fig. 4 are averages over 20 independent experiments. While the training error under the standard BP method steadily reduces, the generalization error increases rapidly. On the other hand, both the AWGN and the LME method avoid overfitting

Authorized licensed use limited to: Zhejiang University. Downloaded on August 21,2020 at 08:33:14 UTC from IEEE Xplore. Restrictions apply.

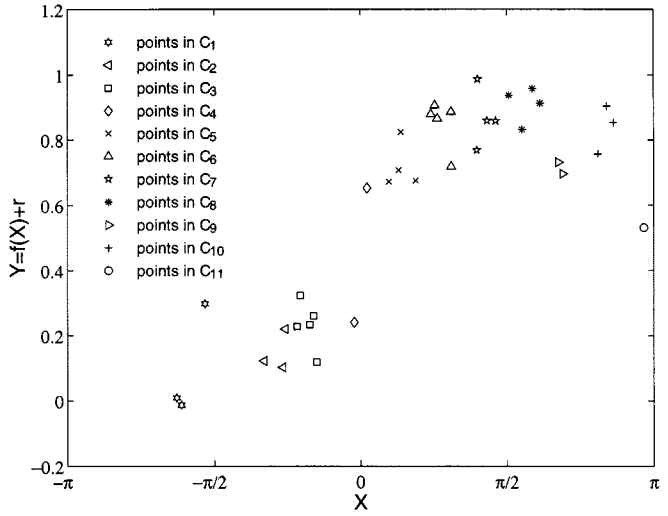


Fig. 1. Data points and $K = 11$ clustering of the training set of Example A.

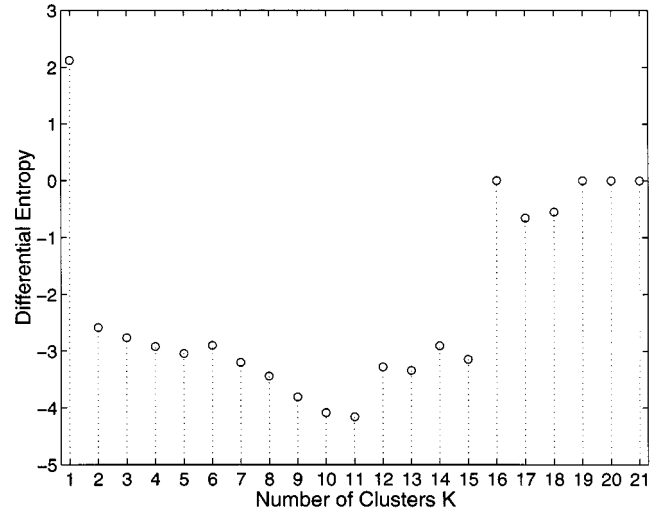


Fig. 2. Differential entropy of the LME pdf versus number of clusters for the data set of Fig. 1.

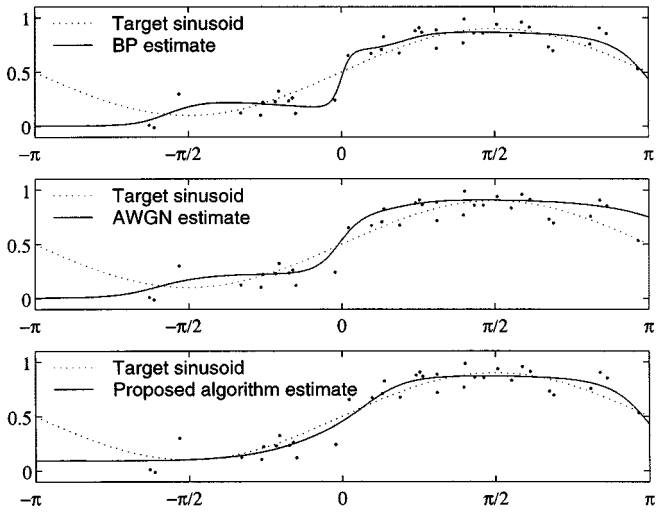


Fig. 3. The target sinusoid function and MLP estimates for the training set of Fig. 1.

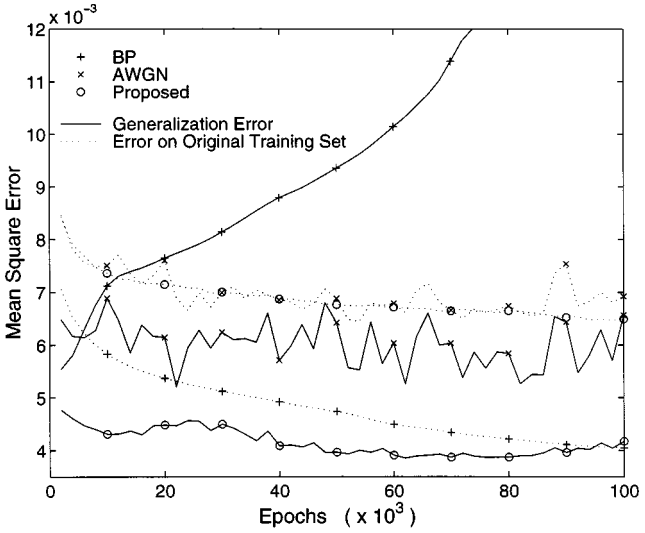


Fig. 4. Generalization error per epoch for the training techniques considered in Example A (the results are averages over 20 independent experiments).

with the latter leading to network designs with superior generalization ability. It is interesting to observe that the LME generalization error is significantly lower than the LME error on the original training set and, in fact, is lower than the standard BP *training error* over as many as 90 K epochs.

B. Multiple-Input/Single-Output Example

The following function was used in [34] in the context of multivariate adaptive regression:

$$f(x_1,\ldots,x_{10}) = 0.1e^{4x_1} + \frac{4}{1+e^{-20(x_2-0.5)}} + 3x_3 + 2x_4 + x_5, \quad [x_1 x_2 \ldots x_{10}]^T \in [0,1]^{10}. \tag{21}$$

This function exhibits nonlinear additive dependence in the first two variables, linear dependence in the next three, and it is totally independent of the last five variables. The ten input variables are generated independently and uniformly in the unit hypercube and zero-mean unit-variance Gaussian "observation" noise is added. The training set consists of $N = 80$ input–output data vectors that we use to train a 10-10-1 network using all three methods of interest, exactly as in Example A. Our studies for AWGN training set expansion indicated significant sensitivity to the selection of the noise variance. In fact, for acceptable performance we had to tune and fix the variance at $\sigma^2 = 10^{-2}$, as opposed to the maximum likelihood cross-validation values prescribed in [20]. The generalization performance of instances of the trained networks was estimated by numerical calculation of the *test error* on 10 000 new random input–output data vectors.

Fig. 5 presents the *test error* and the error on the original 80-point training set for all three networks as a function of the training epochs. The results shown are averages over 20 independent experiments. Standard BP overfits undesirably and AWGN expansion does not avoid overfitting completely. For the LME method, the test error is again lower than the training error.

Authorized licensed use limited to: Zhejiang University. Downloaded on August 21,2020 at 08:33:14 UTC from IEEE Xplore. Restrictions apply.

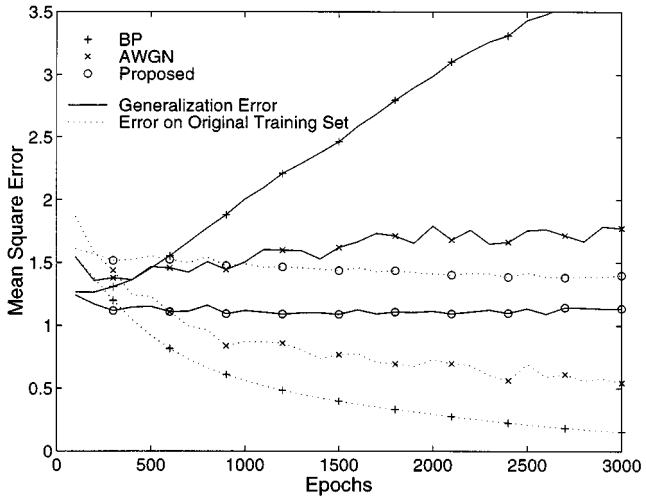


Fig. 5. Generalization error per epoch for Example B (the results are averages over 20 independent experiments).

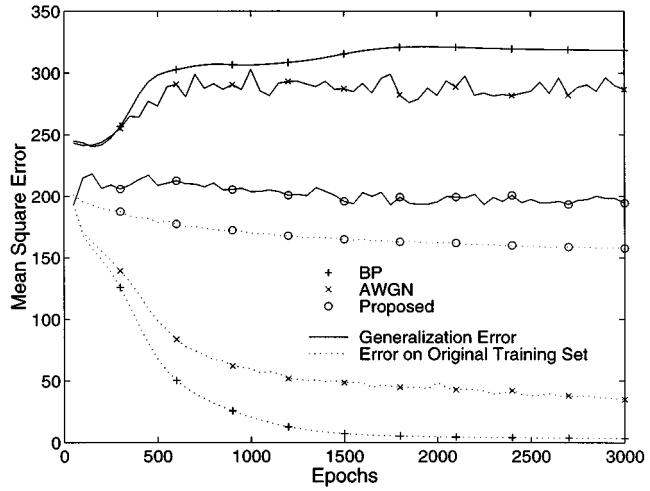


Fig. 6. Generalization error per epoch for Example C (the results are averages over 20 independent random network initializations and training set expansions).

C. Real Data Example

We retrieve quarterly financial data of the Coca-Cola Company, beginning with the first quarter of 1989 and ending with the first quarter of 1997 (a total of 33 consecutive quarters). For each quarter $n = 1, 2, \ldots, 33$ we define an input vector $\mathbf{X}_n \in \mathcal{R}^8$ and an output scalar $Y_n \in \mathcal{R}$. The input vector $\mathbf{X}_n \in \mathcal{R}^8$ is formed by the following eight "ratios" [38], [39]: debt to equity, current ratio, return on common shareholders equity, return on assets, return on equity, quick ratio, profit margin, and asset turnover. The output scalar Y_n is the stock price at the end of the following quarter, that is at the end of the $(n+1)$th quarter.

We consider an 8-25-1 FF network and we attempt to predict the stock market value of one share of the Coca-Cola Company. The input–output pairs $\mathbf{Z}_n = (\mathbf{X}_n, Y_n) \in \mathcal{R}^9$ constitute the 33 available data vectors. The first 19 data vectors (first quarter 1989—third quarter 1994) form the training set and the remaining 14 (fourth quarter 1994—first quarter 1997) form the test set. For AWGN training set expansion acceptable results were obtained by tuning the variance to $\sigma^2 = 10^{-2}$ (as opposed to the variance value 0.0477 suggested by the cross-validation method in [20]). For the LME method we have $K = 2$ and $\sigma_M^2 = 0.062$.

The results presented in Fig. 6 are averages over 20 network training procedures with independent random network initializations. We notice that both the conventional BP and the AWGN expansion method overfit the training set, while LME expansion avoids overfitting and maintains a significantly lower test error.

V. Conclusion

Theoretically, the minimization of the generalization error of a BP-trained MLP requires an infinite sequence of training data drawn from the true joint input–output probability distribution. Given a finite training data set, data recycling—that is training with the same data over and over again—minimizes the error on the exact points of the training set as if all of the joint input–output probability mass were concentrated on these points. In the neural network literature this behavior has been usually referred to as overfitting.

In search of training procedures that may combat overfitting and lead to improved MLP generalization ability, in this work we proposed K-means clustering of the given finite training set. Then, upon establishing the most entropic colored Gaussian pdf estimate per cluster, the overall unconditional joint input–output pdf estimate takes the form of a mixture of colored Gaussians. To optimize the partitioning of the given training set with respect to the number of clusters, we proposed to choose the number of clusters that results to the least entropic Gaussian mixture upon equal-likelihood cross-validated shaping.

The numerical studies of Section IV on synthetic and real data showed significant resistance to overfitting over conventional data-recycling BP training and artificially AWGN infected BP training. In fact, no overfitting was recorded in the sense of an increasing generalization error over extended training periods.

References

[1] P. J. Werbos, "Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences," Ph.D. dissertation, Harvard Univ., Cambridge, MA, Aug. 1974.

[2] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*. Cambridge, MA: MIT Press, 1986, vol. I, ch. 8.

[3] H. White, "Some asymptotic results for learning in single hidden-layer feedforward network models," *J. Amer. Statist. Assoc.*, vol. 84, no. 408, pp. 1003–1013, Dec. 1989.

[4] D. A. Pados and P. Papantoni-Kazakos, "New nonleast-squares neural network learning algorithms for hypothesis testing," *IEEE Trans. Neural Networks*, vol. 6, pp. 596–609, May 1995.

[5] F. Rosenblatt, *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*. Washington, DC: Spartan, 1962.

[6] H. Robbins and S. Monro, "A stochastic approximation method," *Ann. Math. Statist.*, vol. 22, pp. 400–407, 1951.

[7] J. Kiefer and J. Wolfowitz, "Stochastic estimation of the maximum of a regression function," *Ann. Math. Statist.*, vol. 23, pp. 462–466, 1952.

[8] D. J. Sakrison, "Stochastic approximation: A recursive method for solving regression problems," in *Advances in Communication Systems 2*. New York: Academic, 1966, vol. 2, pp. 51–106.

[9] J. R. Blum, "Approximation methods which converge with probability one," *Ann. Math. Statist.*, vol. 25, pp. 382–386, 1954.

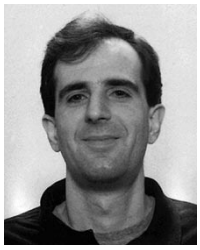
Authorized licensed use limited to: Zhejiang University. Downloaded on August 21,2020 at 08:33:14 UTC from IEEE Xplore. Restrictions apply.

[10] S. J. Farlow, *Self Organizing Methods in Modeling: GMDH-Type Algorithms*. New York: Marcel Dekker, 1984.
[11] M. G. Bello, "Enhanced training algorithms, and integrated training/architecture selection for multilayer perceptron networks," *IEEE Trans. Neural Networks*, vol. 3, pp. 864–875, Nov. 1992.
[12] H. Akaike, "Statistical predictor identification," *Ann. Inst. Statist. Math.*, vol. 22, no. 2, pp. 203–217, 1970.
[13] J. Rissannen, "Modeling by shortest data description," *Automatica*, vol. 14, no. 5, pp. 465–471, Sept. 1978.
[14] G. Schwarz, "Estimating the dimension of a model," *Ann. Statist.*, vol. 6, no. 2, pp. 461–464, Mar. 1977.
[15] A. R. Barron and R. Barron, "Statistical learning networks: A unifying view," in *Proc. 20th Symp. Interface*, E. J. Wegman, D. I. Gantz, and J. J. Miller, Eds., 1989, pp. 192–202.
[16] R. Reed, "Pruning algorithms—A survey," *IEEE Trans. Neural Networks*, vol. 4, pp. 740–747, Sept. 1993.
[17] Y. le Cun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Comput.*, vol. 1, no. 4, pp. 541–551, 1989.
[18] S. M. Peeling, R. K. Moore, and M. J. Tomlinson, "The multilayer perceptron as a tool for speech pattern processing research," in *Proc. IoA Autumn Conf. Speech Hearing*, 1986.
[19] D. C. Plaut, S. J. Nowlan, and G. E. Hinton, "Experiments on learning by backpropagation," Carnegie-Mellon Univ., Pittsburgh, PA, Tech. Rep., 1986.
[20] L. Holmstrom and P. Koistinen, "Using additive noise in backpropagation training," *IEEE Trans. Neural Networks*, vol. 3, pp. 24–38, Jan. 1992.
[21] E. Parzen, "On estimation of a probability density function and mode," *Ann. Math. Statist.*, vol. 33, pp. 1065–1076, 1962.
[22] M. Rosenblatt, "Remarks on some nonparametric estimates of a density function," *Ann. Math. Statist.*, vol. 27, pp. 832–837, 1956.
[23] T. Cacoullos, "Estimation of a multivariate density," *Ann. Inst. Statist. Math.*, vol. 18, no. 2, pp. 179–189, 1966.
[24] C. M. Bishop, "Training with noise is equivalent to Tikhonov regularization," *Neural Comput.*, vol. 7, no. 1, pp. 108–116, 1995.
[25] K. Matsuoka, "Noise injection into inputs in backpropagation learning," *IEEE Trans. Syst., Man, Cybern.*, vol. 22, pp. 436–440, May 1992.
[26] J. Van Ness, "On the dominance of nonparametric Bayes rule discriminant algorithms in high dimensions," *Pattern Recognit.*, vol. 12, no. 6, pp. 355–368, 1980.
[27] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: Wiley, 1991.
[28] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*. New York: McGraw-Hill, 1991.
[29] E. J. Kelly, "An adaptive detection algorithm," *IEEE Trans. Aerospace Electron. Syst.*, vol. AE–22, pp. 115–127, Mar. 1986.
[30] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. R. Statist. Soc.*, vol. 39, pp. 1–38, 1977.
[31] R. P. W. Duin, "On the choice of smoothing parameters for Parzen estimators of probability density functions," *IEEE Trans. Comput.*, vol. 25, pp. 1175–1179, Nov. 1976.
[32] Y. S. Chow, S. Geman, and L. D. Wu, "Consistent cross-validated density estimation," *Ann. Statist.*, vol. 11, no. 1, pp. 25–38, Mar. 1983.
[33] J. S. Marron, "An asymptotically efficient solution to the bandwidth problem of kernel density estimation," *Ann. Statist.*, vol. 13, no. 3, pp. 1011–1023, Sept. 1985.
[34] J. H. Friedman, "Multivariate adaptive regression splines," *Ann. Statist.*, vol. 19, no. 1, pp. 1–141, Mar. 1991.
[35] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*. New York: Wiley, 1973.
[36] J. Makhoul, S. Roucos, and H. Gish, "Vector quantization in speech coding," *Proc. IEEE*, vol. 73, Nov. 1985.
[37] M. R. Anderberg, *Cluster Analysis for Applications*. New York: Academic, 1973.
[38] P. Danos and E. A. Imhoff Jr., *Introduction to Financial Accounting*. Boston, MA: Irwin, 1994.
[39] Z. Bodie, A. Kane, and A. J. Marcus, *Essentials of Investments*. Boston, MA: Irwin, 1995.

George N. Karystinos (S'98) was born in Athens, Greece, on April 12, 1974. He received the Diploma degree in computer engineering and science from the University of Patras, Patras, Greece, in 1997. He is currently pursuing the Ph.D. degree in electrical engineering at the State University of New York, Buffalo.

Since 1998, he has been a Research Assistant with the Communications and Signals Group in the Department of Electrical Engineering at the State University of New York, Buffalo. His research interests are in the areas of wireless multiple access communications, statistical signal processing, and neural networks.

Mr. Karystinos is a student member of the IEEE Communications Society and a member of Eta Kappa Nu.



Dimitris A. Pados (M'95) was born in Athens, Greece, on October 22, 1966. He received the Diploma degree in computer engineering and science from the University of Patras, Patras, Greece, in 1989 and the Ph.D. degree in electrical engineering from the University of Virginia, Charlottesville, in 1994.

He was an Applications Manager in the Digital Systems and Telecommunications Laboratory, Computer Technology Institute, Patras, Greece, from 1989 to 1990. From 1990 to 1994 he was a Research Assistant in the Communications Systems Laboratory, Department of Electrical Engineering, University of Virginia, Charlottesville. From 1994 to 1997 he held an Assistant Professor position in the Department of Electrical and Computer Engineering and the Center for Telecommunications Studies, University of Louisiana, Lafayette. Since August 1997 he has been an Assistant Professor with the Department of Electrical Engineering, State University of New York at Buffalo. His research interests are in the areas of wireless multiple access communications, detection of spread-spectrum signals, adaptive antenna and radar arrays, and neural networks.

Dr. Pados is a member of the IEEE Communications Society and the Communication Theory Technical Committee.

Authorized licensed use limited to: Zhejiang University. Downloaded on August 21,2020 at 08:33:14 UTC from IEEE Xplore. Restrictions apply.