# Mixture of Experts Regression Modeling by Deterministic Annealing

Ajit V. Rao, *Student Member, IEEE,* David Miller, *Member, IEEE,*
Kenneth Rose, *Member, IEEE,* and Allen Gersho, *Fellow, IEEE*

***Abstract*—We propose a new learning algorithm for regression modeling. The method is especially suitable for optimizing neural network structures that are amenable to a statistical description as mixture models. These include mixture of experts, hierarchical mixture of experts (HME), and normalized radial basis functions (NRBF). Unlike recent maximum likelihood (ML) approaches, we directly minimize the (squared) regression error. We use the probabilistic framework as means to define an optimization method that avoids many shallow local minima on the complex cost surface. Our method is based on deterministic annealing (DA), where the entropy of the system is gradually reduced, with the expected regression cost (energy) minimized at each entropy level. The corresponding Lagrangian is the system's "free-energy," and this annealing process is controlled by variation of the Lagrange multiplier, which acts as a "temperature" parameter. The new method consistently and substantially outperformed the competing methods for training NRBF and HME regression functions over a variety of benchmark regression examples.**

***Index Terms*—Deterministic annealing, mixture of experts, neural networks, regression.**

## I. Introduction

In RECENT years, the study of neural networks has been enriched by an infusion of ideas from diverse fields, including statistics and probability theory, information theory, physics, and biology. These ideas have led to reinterpretation of existing network structures; proposals of new network structures; and novel learning algorithms based on optimization techniques, principles, and criteria from these fields. A prime example, which is the focus of the present paper, is the development of neural network models that are inspired by mixture models from statistics [26], [38]. This class includes the structures known as "mixture of experts" [16] and "hierarchical mixture of experts" [17], as well as normalized radial basis functions [28]. We will refer to this class generally as *mixture of experts* (ME) models. ME's have been suggested for a variety of problems, including classification [13], [16], control [15], [17], and regression tasks [17], [39], [40].

The main focus of this paper is the regression problem: Given a training set of input–output pairs $\mathcal{T} \equiv \{(\mathbf{x}_i, \mathbf{y}_i)\}$, where $\mathbf{x}_i \in \mathcal{R}^m$, $\mathbf{y}_i \in \mathcal{R}^n$ are drawn from an unknown underlying distribution, design a mapping $g$: $\mathcal{R}^m \rightarrow \mathcal{R}^n$ that minimizes the expected regression error, which, in the case of squared error, is given by $E[\|\mathbf{y} - g(\mathbf{x})\|^2]$. To formulate the ME model for regression problems, we define the "local expert" regression function $f(\mathbf{x}, \Lambda_j)$, where $\Lambda_j$ is the set of model parameters for local model $j$. Here, $f(\mathbf{x}, \Lambda_j)$ may be constant, linear, polynomial, or some other simple nonlinear function of $\mathbf{x}$. The ME regression function is defined as

$$g(\mathbf{x}) = \sum_j P[j|\mathbf{x}] f(\mathbf{x}, \Lambda_j) \tag{1}$$

where $P[j|\mathbf{x}]$ is a nonnegative weight of association between input $\mathbf{x}$ and expert $j$ that effectively determines the degree to which expert $j$ contributes to the overall model output. In the literature, these weights are often called *gating units* [16]. We further impose $\sum_j P[j|\mathbf{x}] = 1$, which leads to the natural interpretation of the weight of association or gating unit as a probability of association. We restrict ourselves to the important case where $P[j|\mathbf{x}]$ is a parametric function determined by a parameter set $\Theta$. We then obtain the following statistical interpretation of the model. Input–output pair $(\mathbf{x}_i, \mathbf{y}_i)$ is generated by first randomly sampling $\mathbf{x}_i$ according to some input density and then randomly selecting a local model according to the probability mass function $\{P[j|\mathbf{x}_i]\}$. For the chosen model $k$, the output is then generated as a random variable whose mean is $f(\mathbf{x}_i, \Lambda_k)$. From this viewpoint, $g(\mathbf{x})$ in (1) is interpreted as the expected value of the output, given input $\mathbf{x}$. It is important to note the well-known fact that the conditional expectation is the minimum mean-squared error (MMSE) estimator.

There are several additional advantages to the ME structure. One is the fact that ME is an effective compromise between purely local, piecewise models such as classification and regression trees (CART) [1] and "global" models such as the multilayer perceptron (MLP) [37]. By "purely local, piecewise," it is meant that the input space is hard partitioned to regions, each with its own exclusive expert model. Effectively, the piecewise regression function is composed of a patchwork of local regression functions that collectively cover the input

Manuscript received August 14, 1997. This work was supported in part by the National Science Foundation under Grant NCR-9314335, the University of California MICRO program, ACT Networks Inc., Advanced Computer Communications, Cisco Systems, Inc., DSP Group Inc., DSP Software Engineering Inc., Fujitsu Laboratories of America Inc., General Electric Company, Hughes Electronics Corp., Intel Corp., Moseley Associates Inc., National Semiconductor Corp., Nokia Mobile Phones, Qualcomm Inc., Rockwell International Corp., and Texas Instruments Inc. D. Miller was supported by NSF Career Award NSF IRI-9624870. The associate editor coordinating the review of this paper and approving it for publication was Prof. Jenq-Neng Hwang.

A. V. Rao, K. Rose, and A. Gersho are with the Department of Electrical and Computer Engineering, University of California, Santa Barbara, CA 93106 USA.

D. Miller is with the Pennsylvania State University, University Park, PA 16802 USA.

Publisher Item Identifier S 1053-587X(97)08066-5.

space. In addition to partitioning the input space, the model parameter set is partitioned into submodels that are only "active" for a particular local input region. By contrast, in global models such as MLP's, there is a single regression function that must fit the data well everywhere with no explicit partitioning of the input space nor subdivision of the parameter set. One advantage of piecewise solutions lies in the ease of their interpretation—in particular, the role of individual parameters and individual submodels is easily discerned. This is not the case for global models, where it is more difficult to ascertain the role of individual parameters.

The connection between ME models and local piecewise models such as CART is easily seen by noting that piecewise models are the special case of (1), where $P[j|\mathbf{x}]$ is restricted to the values $\{0, 1\}$, i.e., the limiting case of zero randomness. Like the pure piecewise models, the ME structure effectively decomposes the regression problem into learning a set of (expert) models, each of which fits the data well in some local region. However, none has exclusive ownership of a region. In this (somewhat fuzzified) "divide-and-conquer" sense [17], these structures simplify the learning and modeling problem. Moreover, this type of regression fitting generally yields parsimonious solutions, with parameters added only when they are required to improve the fit in a local region. Parsimonious models are known to yield improved generalization.

Although ME models bear similarity to the piecewise models, there are also important differences. Unlike strictly piecewise regression, which produces a function that is discontinuous at region boundaries, the mixture of expert functions is smooth everywhere due to the averaging in (1). Furthermore, the learning methods employed for piecewise regression function design are typically greedy and suboptimal because of the difficulty of jointly optimizing all the model parameters. Learning for mixture of experts, on the other hand, does naturally involve joint optimization of the entire model. In this sense, the ME model is closer to global models such as multilayer perceptrons, where learning is based on backpropagation [37] or other descent methods over the entire parameter set.

The natural learning criterion for regression is the squared-error cost, which is commonly referred to as the regression error. However in [16] and [17], a maximum likelihood (ML) criterion was preferred. This choice was justified by improved performance (even in the sense of squared-error), ease of optimization, certain desirable properties of the solution, and by the applicability of the popular expectation-maximization (EM) algorithm [5] to the design. In this paper, we reason that the superiority of ML methods is mainly due to the complexity of the squared-error cost "surface," which requires more powerful optimization methods than direct gradient descent to ensure good results. Thus, rather than abandon the squared regression error training criterion, we propose a better method for its minimization. Like the ML-based approach, our method capitalizes on a probabilistic description of the ME model. However, we only use this probabilistic framework to develop a powerful optimization method for minimizing the original objective. This method is based on the deterministic annealing approach to clustering [34]–[36] and its extensions.

The rest of this paper is organized as follows. Section II reviews and discusses the basic learning approaches for ME design, with emphasis on the central issues related to the choice of learning criterion. In Section III, we derive the proposed optimization method for the general ME model and specialize it for the NRBF and HME structures. Experimental results presented in Section IV demonstrate the substantial improvements in performance of the DA method over existing methods on real-world and synthetic data sets.

## II. ML versus Squared Error

In the last section, it was noted that for the ME structures in [16] and [17], an ML training criterion was chosen, even though the possibility of training based on the squared-error cost was recognized [16]. We note that while several different criteria may be appropriate, depending on the particular application, the most common, ultimate objective for regression is to minimize the expected squared error between the true output and the output of the approximating function, i.e., $E[\|\mathbf{y} - g(\mathbf{x})\|^2]$, where the expectation is over the joint pdf of input–output pairs $(\mathbf{x}, \mathbf{y})$. In practice, joint statistics are not directly available, and we must instead use finite-length training and test sets that may not fully characterize the joint statistics.

Given a training set $\mathcal{T} \equiv \{(\mathbf{x}_i, \mathbf{y}_i), i = 1, 2, \cdots, N\}$, the squared-error objective is restated as the minimization of

$$\frac{1}{N} \sum_i \left\| \mathbf{y}_i - \sum_j P[j|\mathbf{x}_i] f(\mathbf{x}_i, \Lambda_j) \right\|^2 \tag{2}$$

over the set of model parameters $\Lambda \equiv \{\Lambda_j\}$ and assignment probability parameters $\Theta$. The ML training criterion [16], [17] consists of maximizing

$$\sum_i \log \sum_j \frac{P[j|\mathbf{x}_i]}{(2\pi\sigma^2)^{n/2}} e^{-[\|\mathbf{y}_i - f(\mathbf{x}_i, \Lambda_j)\|^2 / 2\sigma^2]} \tag{3}$$

over $\Lambda$, $\Theta$, and the variance parameter $\sigma^2$. The choice of the ML objective for training was justified from several standpoints in [16]. The authors made a surprising but valid observation that ML training led to better performance in the sense of the squared-error criterion. They further noted that ML training was faster than squared-error training. Finally, it was observed that ML training yielded ME models that could be qualitatively categorized as "competitive," whereas squared-error training led to solutions that were more "cooperative." Competitive models can be understood as ME solutions that more closely resemble local piecewise models than global models. In these, only a few experts are strongly activated for any given input. In cooperative models, on the other hand, the representation is far more distributed, with many experts potentially contributing to a given output. In [16], competitive models were favored based on the advantages of a localized representation. In addition to the justification given in [16], ML-based training is also attractive because it can be realized

by the popular EM algorithm [5], [17]. The EM algorithm has useful convergence properties as described in [17]. It also affords an interesting interpretation to the regression problem by essentially hypothesizing that the data was in fact generated by a local piecewise model but with the partitioning of the inputs to experts considered to be unknown or "missing data." The gating units then measure the expected values of this missing data.

The use of ML estimation and the EM algorithm for the mixture of experts structure has stimulated renewed interest in the learning problem for neural networks, opening up an alternative statistical perspective on neural network training. This approach has been successfully applied in several learning contexts [17], [39], [41]. However, despite these promising results and the justification given in [16], we will reason here that the squared-error cost that directly measures the regression error is a more appropriate training criterion.

We first note the mismatch between ML and squared-error minimization. The likelihood maximization of (3) improves the *individual* fit between output $\mathbf{y}_i$ and each expert $f(\mathbf{x}_i, \Lambda_j)$ rather than the cooperative fit based on the ME output $g(\mathbf{x}_i)$. Although this approach encourages each expert *individually* to fit the data well in some localized region, there appear to be no guarantees on the performance of the resulting overall model.

Moreover, we argue that the best regression function is the one that minimizes the regression error, regardless of whether the resulting solution is qualitatively competitive or cooperative. In fact, an important advantage of the ME model seems to be that it admits both possibilities. Thus, if the learning algorithm is successful in minimizing the cost, it should be able to seek either competitive or cooperative results, depending on which provides a better fit to the given data.

It is important to note that by adopting the squared-error criterion, we do not discard the probabilistic interpretation of the model parameters. The weight $P[j|\mathbf{x}]$ is indeed interpreted as the probability of associating input $\mathbf{x}$ with model $j$. The function $f(\mathbf{x}, \Lambda_j)$ is interpreted as the conditional expectation of output $\mathbf{Y}$ given that input $\mathbf{x}$ is assigned to model $j$. However, the training of these parameters is performed to minimize the regression error directly rather than maximize the likelihood objective.

We note that in the closely related problem of pattern classification, there has been a renewed research interest [7], [11], [18], [21] in the optimization of the true, yet complex, cost—misclassification probability—rather than a mismatched but simpler cost function. This approach has found applications in various fields, particularly in speech recognition [19], [29].

At this point, we must reconcile our argument with the finding in [16] that solutions obtained by ML learning are superior to those trained directly for the squared-error cost. In fact, with some qualification, our results are not inconsistent with this finding. More concretely, we have found that ascent on the likelihood cost surface sometimes leads to better solutions in the sense of squared error than those obtained by direct gradient descent on the squared-error cost surface itself. However, what this primarily suggests is that the squared-error surface may be more complex than the ML surface, with numerous poor local optima to trap descent methods. Thus, rather than abandon the squared-error training criterion, our proposed line of attack is to seek a better method for its minimization.

## III. Deterministic Annealing

In recent years, optimization methods grounded in an analogy to physical and chemical processes have been actively developed to tackle combinatorial optimization problems such as the traveling salesman problem [8]. An important stochastic method known as simulated annealing [20] is a general optimization technique that converges to the globally optimal solution in probability. However, the computational complexity of an implementation assuring this convergence often exceeds what can be practically realized. The learning method that we develop here builds on recent approaches that capture some of the power of the stochastic annealing optimization method while reducing computational complexity via a deterministic approximation. Several related methods have been described as "deterministic annealing" and "mean-field annealing" and have been developed in different fields. Our approach builds on the deterministic annealing (DA) approach for data clustering and related problems [34]–[36] and its extension to incorporate structural constraints on the data assignments [27] with particular emphasis on the problems of statistical classification [27] and piecewise regression [33]. In all the above problems, where DA has already been used successfully, the common goal was the design of a system that implements hard assignment of data to groups or classes. The DA method introduced randomization within the design phase in order to allow global optimization over probabilities, ultimately leading to hard assignments as the "temperature" is lowered to zero.

The mixture of experts regression model bears some similarity to piecewise regression as data is assigned to local models. However, an important difference in the problem definition is that each data point is associated in probability with the various local models. Hence, randomized association is inherent to the model and does not have to be introduced artificially. We next derive the deterministic annealing approach for the design of a general mixture model, followed by specialization to develop the DA method for the NRBF and the HME regression architectures.

### A. DA Design Method for a General Mixture of Experts Model

Fundamentally, we view the ME design problem as the problem of optimization of the data assignment rule that governs the relation between data and local models. However, unlike hard partitional clustering problems, each data point is associated *in probability* to the local models. In other words, ME model design does not impose hard data associations but, rather, seeks the optimal probabilistic assignments $\{P[j|\mathbf{x}_i]\}$ (as well as the model parameter set $\Lambda$) that minimize the squared-error cost

$$D = \frac{1}{N} \sum_i \left\| \mathbf{y}_i - \sum_j P[j|\mathbf{x}_i] f(\mathbf{x}_i, \Lambda_j) \right\|^2 \qquad (4)$$

where $\{P[j|\mathbf{x}_i]\}$ are determined by the parameter set $\Theta$, as defined for the specific ME structure. The Shannon entropy of the association between the data and local models is

$$H = -\frac{1}{N}\sum_i \sum_j P[j|\mathbf{x}_i] \log P[j|\mathbf{x}_i]. \tag{5}$$

The entropy may be viewed as a measure of the randomness of the probabilistic assignments. The ultimate objective is the optimization of the probabilities and model parameters to minimize $D$ whose cost surface is typically riddled with poor local minima. In this work, we propose to apply an "annealing" process, whereby a high level of randomness (entropy) is imposed on the system, and then, the constraint is gradually reduced. The basic constrained optimization problem is therefore

$$\min_{\Theta,\Lambda} D \text{ subject to } H = H_0 \tag{6}$$

where $H_0$ is the imposed level of randomness. Effectively, this optimization seeks the best randomized regression model, given a prescribed level of randomness $H_0$. The annealing process involves solving a sequence of optimizations of this type for decreasing values of $H_0$. The constrained optimization is, of course, equivalent to minimization of the Lagrangian

$$F = D - TH, \tag{7}$$

where $T$ is the Lagrange multiplier. It is important to note that the quantity $F$ can also be identified as the Helmholtz free energy of a system with "energy" $D$, entropy $H$, and "temperature," $T$. Thus, the annealing process involves minimizing $F$ starting from high $T$ and tracking the minimum for a sequence of decreasing values of $T$. At high $T$, the objective is, in fact, entropy maximization, which is achieved by the uniform distribution. As $T$ is lowered, increasing emphasis is placed on minimizing $D$, which also has the effect of reducing the entropy. At $T = 0$, we seek to minimize $D$ regardless of the level of entropy, which is precisely the ultimate objective. The annealing process helps to avoid shallow local minima, as will be demonstrated in the results section.

We can gain some intuition about this annealing process by noting that solutions with high entropy can be characterized as highly "cooperative," whereas solutions with low entropy are more "competitive." Thus, the annealing process effectively conducts a search for the best regression model, starting with the constraint of a high degree of cooperation, and gradually relaxing this constraint. Since at $T = 0$ there is no constraint on the entropy, the method ultimately seeks the best regression solution, regardless of whether the result is "competitive" or "cooperative." Note that at a very high temperature ($T \to \infty$), the uniform distribution implies that all the local models are identical. The effective model size (number of nondistinct local models) is one. As $T$ is lowered, more emphasis is placed on reducing the regression error, thereby leading to a gradual growth in the effective model size. The entropy constrained formulation, however, ensures that the model size will increase only if the improvement in the regression error warrants the decrease in the entropy of the associations.

In practice, the minimization of $F$ is achieved by a series of gradient descent steps on this cost at each temperature $T$. An "annealing schedule" $q(T)$ determines the procedure for gradually cooling the system. When the system has reached thermal equilibrium at a temperature $T$, the temperature update $T \leftarrow q(T)$ is applied followed by minimization of $F$ at the new temperature. An exponential schedule $q(T) = \alpha T$, where $\alpha < 1$ worked well in all our experiments.

The DA algorithm can be summarized as follows.

1) Set parameters: initial temperature $T_i$, final temperature, $T_f$, and annealing schedule function $q(\cdot)$.
2) Set $T = T_i$.
3) Minimize $F = D - TH$ over $(\Theta, \Lambda)$.
4) Lower temperature: $T \leftarrow q(T)$.
5) If $T > T_f$, go to Step 3.

Although any standard local optimization method can be used to minimize the free energy in Step 3, we used a simple gradient descent method in our experiments.

The DA design approach described in this section is quite general and can be specialized to any specific mixture of experts model. Different ME structures simply correspond to different parametric forms for the association probabilities $\{P[j|\mathbf{x}]\}$ and the local models $\{f(\mathbf{x}, \Lambda_j)\}$. Hence, the corresponding DA design methods differ only in the gradient step prescription for the free-energy minimization of Step 3. We next consider two important ME models—the normalized radial basis function (NRBF) and the hierarchical mixture of experts (HME) and rederive the DA design method for these structures.

### *B. Normalized Radial Basis Function (NRBF)*

The radial basis function (RBF) architecture is an important class of neural networks. Typically, the RBF network has two stages. In the lower (first) stage, the "activation" of each node is determined by a set of RBF's. In the second stage, the activations are combined linearly to obtain the regression estimate. Although there are many possible choices of RBF's, perhaps the most important and commonly used are the *Gaussian basis functions*

$$R_k(\mathbf{x}) = e^{-(\|\mathbf{x}-\mathbf{m}_k\|^2/2\sigma^2)}. \tag{8}$$

The vectors $\mathbf{m}_k$ are the "centers" or "prototype vectors," and $\sigma$ is the "bandwidth." The RBF was suggested for general interpolation problems [32] and used in the context of neural networks [2], [28]. RBF's have some useful properties that make them particularly attractive for regression applications [9], [12]. They have been used successfully in a wide variety of practical applications in regression [3], [6], [30], [31] as well as in classification [22], [23].

An important extension of the basic RBF architecture is the normalized RBF (NRBF) shown in Fig. 1. The NRBF architecture is organized in two layers. In the lower layer, we compute the hidden outputs via the normalized RBF's

$$P[k|\mathbf{x}] = \frac{R_k(\mathbf{x})}{\sum_{k'} R_{k'}(\mathbf{x})}. \tag{9}$$

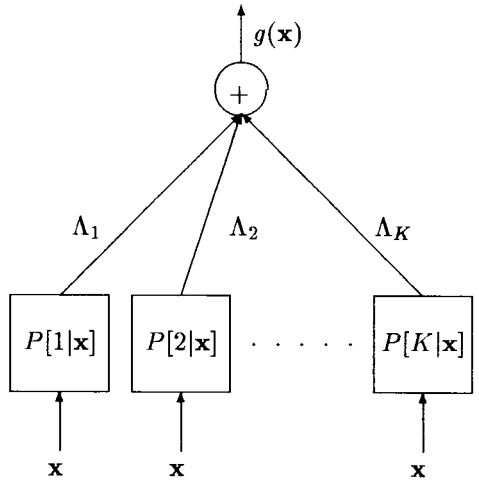


Fig. 1. Normalized radial basis function (NRBF) architecture.

The second layer then performs the linear operation

$$g(\mathbf{x}) = \sum_{k=1}^{K} P[k|\mathbf{x}]\Lambda_k. \tag{10}$$

This architecture may be interpreted as an ME model where the weights $\{P[k|\mathbf{x}]\}$ represent the probabilities of association with the corresponding constant local models $\{\Lambda_k\}$. Further, these probabilities are determined by the parameters $\Theta \equiv \{\{\mathbf{m}_k\}, \sigma\}$. We wish to optimize the parameter set $\Theta$ jointly with the local model parameter set $\Lambda$ to minimize the regression error

$$D = \frac{1}{N}\sum_{i=1}^{N} ||\mathbf{y}_i - g(\mathbf{x}_i)||^2. \tag{11}$$

One common NRBF design approach was suggested in [28]: Fix the RBF centers $\{\mathbf{m}_k\}$ via a clustering algorithm [24], and then, optimize $\Lambda$ and $\sigma$ to minimize the cost function $D$. A more powerful version of this algorithm optimizes the entire NRBF parameter set $\{\Theta, \Lambda\}$ in the second step. This algorithm is quick but suboptimal, primarily due to the aforementioned difficulties with gradient descent on the cost surface.

We propose the DA approach that avoids many poor local minima on the cost surface. The basic DA optimization step is the minimization of the regression error at a given level of entropy (5) or, equivalently, the minimization of the free energy $F = D - TH$. This free energy minimization is carried out for a sequence of decreasing temperatures ending at $T = 0$. At each temperature $T$, the minimum satisfies the following conditions.

For the RBF centers

$$\frac{\partial F}{\partial \mathbf{m}_k} = \frac{1}{N\sigma^2}\sum_i (\mathbf{x}_i - \mathbf{m}_k)P[k|\mathbf{x}_i][\theta_k(\mathbf{x}_i) - \overline{\theta}(\mathbf{x}_i)] = 0 \tag{12}$$

for the bandwidth parameter

$$\frac{\partial F}{\partial \sigma} = \frac{1}{N\sigma^3}\sum_i \sum_k ||\mathbf{x}_i - \mathbf{m}_k||^2 P[k|\mathbf{x}_i][\theta_k(\mathbf{x}_i) - \overline{\theta}(\mathbf{x}_i)] = 0 \tag{13}$$

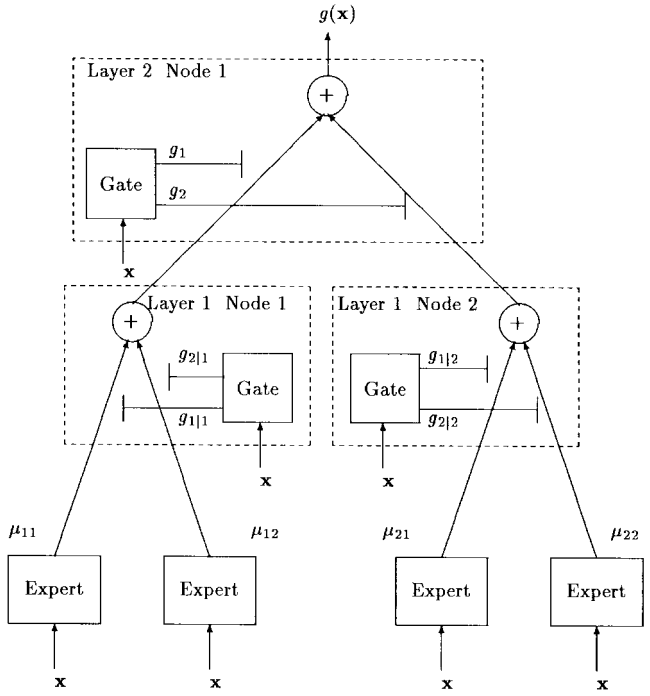


Fig. 2. Two-level binary tree representing the HME architecture for regression.

and for the local model $\Lambda_k$

$$\frac{\partial F}{\partial \Lambda_k} = \frac{2}{N}\sum_i P[k|\mathbf{x}_i][g(\mathbf{x}_i) - \mathbf{y}_i] = 0. \tag{14}$$

In the above equations, we make use of the quantities

$$\theta_k(\mathbf{x}) = 2[g(\mathbf{x}) - \mathbf{y}]\Lambda_k - \frac{T}{2\sigma^2}||\mathbf{x} - \mathbf{m}_k||^2 \tag{15}$$

and their average over the models

$$\overline{\theta}(\mathbf{x}) = \sum_{k=1}^{K} P[k|\mathbf{x}]\theta_k(\mathbf{x}). \tag{16}$$

The gradient expressions above can be viewed as perceptron-like learning rules. For example, a gradient descent step for the prototypes based on (12) can be interpreted as a rule that moves a prototype toward or away from data points, depending on whether their contribution to the cost increases or decreases by association with this prototype. The rules for the $\sigma$ and $\Lambda_k$ can be interpreted in a similar manner.

### *C. Hierarchical Mixture of Experts (HME)*

In its most general form, the hierarchical mixture of experts (HME) is organized as a multilevel, multibranch tree. Although our design method is applicable to this general structure, for simplicity of presentation, we will restrict discussion to the simple two-level, binary-tree HME architecture of Fig. 2.

The leaves of the tree represent simple local regression models (experts). Starting from the root node, we imagine choosing a random branch, recursively, until we arrive at one of the leaves. The conditional distribution for choosing the branches given a node is computed at that node by a "gate."

Specifically, the gate at the root node observes $\mathbf{x}$ and computes the conditional distribution[1]

$$g_j = \frac{e^{\mathbf{v}_j^T \mathbf{x}}}{\sum_m e^{\mathbf{v}_m^T \mathbf{x}}}. \tag{17}$$

Similarly, at node $j$ in the lower layer, the gate computes the conditional distribution

$$g_{k|j} = \frac{e^{\mathbf{v}_{jk}^T \mathbf{x}}}{\sum_l e^{\mathbf{v}_{jl}^T \mathbf{x}}}. \tag{18}$$

One may interpret the hierarchy as a soft tree-structured partition of the input space, based on weight vectors $\{\mathbf{v}_j\}$ and $\{\mathbf{v}_{jk}\}$.

The conditional distribution over the branches induces a distribution over the local models. Specifically, the probability of choosing model $\mu_{jk}$ is given by $p_{jk} \equiv g_j g_{k|j}$. From the ME viewpoint, we are interested in the weighted average of the outputs, i.e., the expectation

$$\begin{aligned} g(\mathbf{x}) &= \sum_{j,k} p_{jk}\mu_{jk} = \sum_j g_j \left\{ \sum_k g_{k|j}\mu_{jk} \right\} \\ &= \sum_j g_j \mu_j \end{aligned} \tag{19}$$

where $\mu_j$ is defined as the term in parenthesis. The straightforward way to compute $g(\mathbf{x})$ is via the architecture in Fig. 2. We propagate the estimates provided by the experts by linearly combining them as we proceed from the leaves to the root node, where the final regression estimate $g(\mathbf{x})$ is produced.

The HME design objective is the optimization of association probability parameters $\Theta \equiv \{\{\mathbf{v}_{jk}\}, \{\mathbf{v}_j\}\}$ and the model parameter set $\Lambda$ to minimize the regression error

$$D = \frac{1}{N} \sum_{i=1}^{N} ||\mathbf{y}_i - g(\mathbf{x}_i)||^2. \tag{20}$$

We are interested in simultaneously controlling the entropy of association given by

$$H = -\frac{1}{N} \sum_i \sum_j \sum_k p_{jk}(\mathbf{x}_i) \log p_{jk}(\mathbf{x}_i). \tag{21}$$

Equivalently, we minimize the Lagrangian $F = D - TH$ at a fixed temperature $T$. As in NRBF design, we use a gradient descent method for the optimization. The free-energy minimization is repeated for a sequence of decreasing temperatures. An important advantage of this approach for the tree architecture is that the gradients can be computed efficiently via a backpropagation-like rule that follows from the chain rule of derivatives. Here, we only write the simpler optimality conditions for the gradients in the case of the two-level hierarchy. Generalization to larger trees is straightforward. The optimality conditions are

$$\frac{\partial F}{\partial \mathbf{v}_j} = \frac{1}{N} \sum_i g_j(\mathbf{x}_i)[\phi_j(\mathbf{x}_i) - \overline{\phi}(\mathbf{x}_i)]\mathbf{x}_i = 0 \tag{22}$$

$$\frac{\partial F}{\partial \mathbf{v}_{jk}} = \frac{1}{N} \sum_i p_{jk}(\mathbf{x}_i)[\phi_{jk}(\mathbf{x}_i) - \overline{\phi}_j(\mathbf{x}_i)]\mathbf{x}_i = 0 \tag{23}$$

and

$$\frac{\partial F}{\partial \mu_{jk}} = \frac{2}{N} \sum_i [g(\mathbf{x}_i) - \mathbf{y}_i] p_{jk}(\mathbf{x}_i) = 0. \tag{24}$$

In the equations above, we have made use of the following additional variables, each associated with a branch in the tree.

$$\phi_j(\mathbf{x}) = 2[g(\mathbf{x}) - \mathbf{y}]\mu_j(\mathbf{x}) + T[\mathbf{v}_j^T \mathbf{x} - h_j(\mathbf{x})] \tag{25}$$

and

$$\phi_{jk}(\mathbf{x}) = 2[g(\mathbf{x}) - \mathbf{y}]\mu_{jk}(\mathbf{x}) + T\mathbf{v}_{jk}^T \mathbf{x} \tag{26}$$

as well as their average values, which have been computed over branches that terminate at the same node:

$$\overline{\phi}(\mathbf{x}) = \sum_j g_j(\mathbf{x})\phi_j(\mathbf{x}) \tag{27}$$

and

$$\overline{\phi}_j(\mathbf{x}) = \sum_k g_{k|j}(\mathbf{x})\phi_{jk}(\mathbf{x}). \tag{28}$$

The quantity $h_j$ denotes the conditional entropy

$$h_j = -\sum_k g_{k|j} \log g_{k|j}. \tag{29}$$

The above expressions for the gradients offer interesting perceptron-like interpretations to the gradient-descent algorithm. Viewing the $\phi$ variables as the contribution of each branch to the cost function, a gradient-descent method based on (22) and (23) may be interpreted as a perceptron-like rule to strengthen (weaken) the association of an input with branches that contribute a cost that is smaller (higher) than the average over all branches that terminate at the same node.

## IV. Results

In this section, we report the results of our experiments comparing the deterministic annealing approach with conventional design methods for NRBF and HME regression functions. The experiments are performed over some popular benchmark data sets from the regression literature. Among these data sets, the first three are real-world applications of regression drawn from the StatLib data set archive[2], whereas the others have been synthetically generated.

In each experiment, we compare the average squared-error obtained over the training set using the DA design method and the alternative design methods. The comparisons are repeated for different network sizes. The network size $K$ refers to the number of local experts used in the mixture model. For the case of binary HME trees with $l$ levels, $K = 2^l$, and for the case of NRBF regression functions, $K$ is the number of

[1] Note that although the variables $\mu_{jk}$, $g_j$, $g_{k|j}$, $h_j$, $p_{jk}$ depend on $\mathbf{x}$, for the sake of notational simplicity, we drop the argument $\mathbf{x}$. Hence, e.g., $\mu_{jk} \equiv \mu_{jk}(\mathbf{x})$ unless otherwise stated.

[2] The StatLib data set archive is accessible on the World-Wide Web at http://lib.stat.cmu.edu/datasets/

TABLE I
COMPARISON OF REGRESSION ERROR OBTAINED USING DA AND GD ALGORITHMS FOR NRBF DESIGN FOR THE BOSTON HOME VALUE PROBLEM. $K$ IS THE NUMBER OF GAUSSIAN BASIS FUNCTIONS

| K | DA | GD |
|---|---|---|
| 1 | 87.7 | 87.7 |
| 2 | 19.7 | 23.8 |
| 4 | 12.9 | 19.3 |
| 6 | 12.6 | 15.7 |
| 10 | 6.5 | 13.7 |

Gaussian basis functions used. Following the most common implementation, the local models are constant functions in the NRBF case and linear functions in the HME case. The alternative design approaches used for comparing our HME design algorithm are

- "GD," which is a gradient descent algorithm to simultaneously optimize all HME parameters for the squared-error cost;
- "ML," which is Jordan and Jacobs's ML approach [17].

For the NRBF regression function, we have compared the DA design approach with the gradient descent algorithm, which is an enhanced version of the method suggested in [28] (as described in the previous section).

In our implementation of the DA algorithm for both NRBF and HME design, we adopt an exponential temperature schedule $q(T) = \alpha T$ with $\alpha = 0.98$. Further, the free-energy minimization at a fixed temperature is performed via a sequence of gradient descent steps. Convergence is determined by comparing the fractional improvement[3] in free energy to a small threshold value $\epsilon = 10^{-4}$.

In our implementation of the GD algorithm for NRBF design, we randomly initialize all parameters, apply the $k$-means algorithm [24] to place the RBF centers, and execute a sequence of gradient descent steps on all parameters. In the GD algorithm for HME function design, a random initialization of all parameters is used. The GD algorithms for both architectures terminate when the fractional improvement is smaller than the threshold, $\epsilon$.

In our implementations of all the above methods, we used an identical improvement threshold ($\epsilon$) to ensure fairness of comparison. To implement the ML approach to design HME functions, we used the algorithm based on iterated recursive least squares (IRLS), which was suggested in [17]. Starting from a random initialization of all parameters, we allow 100 epochs of this recursive algorithm for the solution to converge.

For fair comparison, we take a conservative (worst-case) estimate that the complexity of the DA approach is ten times greater than that of the competing methods. To compensate for the complexity, we allow each competing method to generate results based on ten different random initializations, with the best result obtained among those runs selected for comparison with the DA result. Since the regression function obtained by DA is generally independent of initialization, a single DA run sufficed.

[3]Fractional improvement of a cost function is the ratio between the improvement in the cost resulting from an iteration and the absolute value of the cost before the iteration.

TABLE II
COMPARISON OF REGRESSION ERROR OBTAINED USING DA, GD, AND ML ALGORITHMS FOR HME FUNCTION DESIGN FOR THE BOSTON HOME VALUE PROBLEM. $K$ IS THE NUMBER OF LEAVES IN THE BINARY TREE

| K | DA | GD | ML |
|---|---|---|---|
| 4 | 5.7 | 5.9 | 7.5 |
| 8 | 3.4 | 3.6 | 5.6 |

TABLE III
COMPARISON OF REGRESSION ERROR OBTAINED USING DA AND GD ALGORITHMS FOR NRBF DESIGN FOR THE MORTALITY RATE PREDICTION PROBLEM. $K$ IS THE NUMBER OF GAUSSIAN BASIS FUNCTIONS

| K | DA | GD |
|---|---|---|
| 1 | 3805.1 | 3805.1 |
| 2 | 1148.8 | 2154.0 |
| 4 | 720.8 | 1256.8 |
| 6 | 439.1 | 566.5 |
| 8 | 299.6 | 564.5 |
| 10 | 261.4 | 438.2 |

First, we considered the Boston home value prediction problem [10]. Here, we use data from 506 homes in the Boston area to predict the median price of each home from 13 features that are believed to have some influence on it. Since the features have different dynamic ranges, we first normalized each one to unit variance. Using the entire data for training, we designed NRBF and HME regression functions using DA and alternative methods. Our results in Tables I and II demonstrate that for both mixture models, the DA approach achieves a significantly smaller regression error compared with the other approaches over a variety of network sizes.

Our second data set, which is taken from the environmental sciences, has been used by numerous researchers since its introduction [25] in the early 1970's. Here, we consider the problem of predicting the age-adjusted mortality rate per 100 000 people in a locality from 15 factors that may have possibly influenced it. Since there is data for only 60 localities, we used the entire set for training. Tables III and IV show that for both the NRBF and HME regression structures, over the entire range of network sizes, the DA design approach significantly improved performance over the competing design methods.

The third regression data set is drawn from an application in the food sciences. The problem is that of efficient estimation of the fat content of a sample of meat. (Techniques of analytical chemistry can be used to measure this quantity directly, but it is a slow and time-consuming process.) The data set of measurements was obtained by the Tecator Infratec Food and Feed Analyzer, which estimates the absorption of electromagnetic waves in 100 different frequency bands and

TABLE IV
COMPARISON OF REGRESSION ERROR OBTAINED USING DA, GD, AND ML ALGORITHMS FOR HME DESIGN FOR THE MORTALITY RATE PREDICTION PROBLEM. $K$ IS THE NUMBER OF LEAVES IN THE BINARY TREE

| K | DA | GD | ML |
|---|---|---|---|
| 4 | 18.2 | 121.8 | 70.4 |
| 8 | 2.1 | 12.3 | 41.8 |

TABLE V
COMPARISON OF REGRESSION ERROR OBTAINED USING DA AND GD ALGORITHMS FOR NRBF DESIGN FOR THE FAT CONTENT PREDICTION PROBLEM. $K$ IS THE NUMBER OF GAUSSIAN BASIS FUNCTIONS. "TR" AND "TE" REFER TO TRAINING AND TEST SETS, RESPECTIVELY

| | DA | | GD | |
|---|---|---|---|---|
| K | TR | TE | TR | TE |
| 1 | 159.9 | 168.2 | 159.9 | 168.2 |
| 2 | 52.9 | 58.8 | 131.4 | 159.7 |
| 4 | 28.6 | 32.9 | 119.8 | 138.0 |
| 6 | 27.3 | 40.1 | 74.9 | 83.7 |

TABLE VI
COMPARISON OF REGRESSION ERROR OBTAINED USING DA, GD, AND ML ALGORITHMS FOR HME FUNCTION DESIGN FOR THE FAT CONTENT PREDICTION PROBLEM. $K$ IS THE NUMBER OF LEAVES IN THE BINARY TREE. "TR" AND "TE" REFER TO TRAINING AND TEST SETS, RESPECTIVELY

| | DA | | GD | | ML | |
|---|---|---|---|---|---|---|
| K | TR | TE | TR | TE | TR | TE |
| 4 | 8.3 | 11.5 | 14.1 | 18.1 | 15.1 | 23.9 |
| 8 | 6.9 | 9.8 | 12.8 | 17.2 | 12.5 | 39.7 |

the corresponding fat content as determined by analytical chemistry. As suggested by the data set providers, we divided the data into a training set of size 173 and a test set of size 43. Next, we designed the NRBF and HME regression functions using the DA and conventional design methods for different network sizes. In Tables V and VI, we compare the average squared error obtained over the training and test sets. Again, the DA design approach significantly outperforms the conventional design methods over both training and test sets for both HME and NRBF architectures. Note that allowing the ML approach to use a larger network size does not necessarily improve the test set performance, although performance on the training set improves marginally.

The last set of experiments is based on synthetically generated data. Here, $\mathbf{X} = (x_0, x_1)$ is 2-D, and the training set is generated according to a uniform distribution in the unit square. The output $\mathbf{Y}$ is scalar. We created five different data sets based on the functions $[f_1(), f_2(), \cdots, f_5()]$ specified in [4] and [14]. Each function was used to generate both a training set and test set of size 225. We designed NRBF and HME regression estimates for each data sets using both DA and the competitive design approaches. The results shown in

TABLE VII
COMPARISON OF REGRESSION ERROR OBTAINED USING DA AND GD ALGORITHMS FOR NRBF DESIGN TO APPROXIMATE FUNCTIONS, $f_1() \cdots f_5()$. $K$ IS THE NUMBER OF GAUSSIAN BASIS FUNCTIONS. "TR" AND "TE" REFER TO TRAINING AND TEST SETS, RESPECTIVELY

| Method | K | $f_1()$ | $f_2()$ | $f_3()$ | $f_4()$ | $f_5()$ |
|---|---|---|---|---|---|---|
| DA(TR) | 8 | 0.001 | 0.008 | 0.01 | 0.08 | 0.13 |
| DA(TE) | 8 | 0.001 | 0.009 | 0.01 | 0.09 | 0.13 |
| GD(TR) | 8 | 0.02 | 0.044 | 0.16 | 0.19 | 0.24 |
| GD(TE) | 8 | 0.02 | 0.049 | 0.17 | 0.17 | 0.23 |
| DA(TR) | 16 | 0.001 | 0.003 | 0.01 | 0.05 | 0.02 |
| DA(TE) | 16 | 0.001 | 0.005 | 0.01 | 0.05 | 0.03 |
| GD(tr) | 16 | 0.02 | 0.012 | 0.14 | 0.06 | 0.24 |
| GD(te) | 16 | 0.02 | 0.017 | 0.12 | 0.07 | 0.23 |

TABLE VIII
COMPARISON OF REGRESSION ERROR OBTAINED USING DA, GD, AND ML ALGORITHMS FOR HME FUNCTION DESIGN TO APPROXIMATE FUNCTIONS $f_1() \cdots f_5()$. $K$ IS THE NUMBER OF LEAVES IN THE BINARY TREE. "TR" AND "TE" REFER TO TRAINING AND TEST SETS, RESPECTIVELY

| Method | K | $f_1()$ | $f_2()$ | $f_3()$ | $f_4()$ | $f_5()$ |
|---|---|---|---|---|---|---|
| DA(TR) | 4 | 0.0006 | 0.02 | 0.18 | 0.20 | 0.19 |
| DA(TE) | 4 | 0.0006 | 0.02 | 0.18 | 0.25 | 0.21 |
| GD(TR) | 4 | 0.0079 | 0.06 | 0.39 | 0.36 | 0.35 |
| GD(TE) | 4 | 0.0082 | 0.06 | 0.47 | 0.43 | 0.38 |
| ML(TR) | 4 | 0.026 | 0.08 | 0.86 | 0.36 | 0.43 |
| ML(TE) | 4 | 0.039 | 0.12 | 0.79 | 0.46 | 0.51 |
| DA(TR) | 8 | 0.0003 | 0.01 | 0.09 | 0.08 | 0.17 |
| DA(TE) | 8 | 0.0003 | 0.02 | 0.09 | 0.01 | 0.16 |
| GD(TR) | 8 | 0.0063 | 0.05 | 0.12 | 0.35 | 0.28 |
| GD(TE) | 8 | 0.0079 | 0.05 | 0.12 | 0.40 | 0.30 |
| ML(TR) | 8 | 0.011 | 0.03 | 0.12 | 0.09 | 0.32 |
| ML(TE) | 8 | 0.016 | 0.04 | 0.14 | 0.14 | 0.44 |

Tables VII and VIII show improved performance of the DA method that is consistent with the results obtained for the other benchmark sets.

Although, we demonstrate significant improvements in regression performance using the DA design approach, this gain is obtained at the expense of an increase in complexity. In our experiments, the increase in complexity is by a factor of 2–10.

## V. CONCLUSIONS

We have presented an annealing approach for the design of regression models based on the mixture of expert architectures. This class includes the recent hierarchical mixtures of experts [17] as well as normalized radial basis functions

[28]. There has been much recent interest in these structures, prompted mostly by new learning algorithms that emphasize a probabilistic description of the model and redefine the learning problem from a statistical perspective as ML estimation. Although these algorithms have several attractive properties, including efficient learning based on the EM algorithm, we have identified two shortcomings, namely, mismatch between the design objective and the regression error and susceptibility of design methods to poor local minimum traps. The proposed DA method capitalizes on the probabilistic model description to directly attack the regression error minimization criterion while avoiding many shallow local optima of the cost. Experimental results provide ample evidence of the superior performance of the DA method.

## ACKNOWLEDGMENT

The authors would like to thank Prof. M. I. Jordan and Prof. R. A. Jacobs for providing their software for the purpose of the comparisons in this paper.

## REFERENCES

[1] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees.* Belmont, CA: Wadsworth, 1984.
[2] D. S. Broomhead and D. Lowe, "Multivariable functional interpolation and adaptive networks," *Complex Syst.*, vol. 2, no. 3, pp. 321–355, June 1988.
[3] S. Chen, S. A. Billings, C. F. N. Cowan, and P. M. Grant, "Nonlinear systems identification using radial basis functions," *Int. J. Syst. Sci.*, vol. 21, no. 12, pp. 2513–2539, Dec. 1990.
[4] V. Cherkassky, Y. Lee, and H. Lari-Najafi, "Self-organizing network for regression: Efficient implementation and comparative evaluation," in *Int. Joint Conf. Neural Networks*, 1991, vol. 1, pp. 79–84.
[5] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum-likelihood from incomplete data via the EM algorithm," *J. R. Stat. Soc., B*, vol. 39, pp. 1–38, 1977.
[6] F. Diaz-de-Maria and A. R. Figueiras-Vidal, "Nonlinear prediction for speech coding using radial basis functions," in *Proc. ICASSP*, vol. 1, pp. 788–791, 1995.
[7] H. Do-Tu and M. Installe, "Learning algorithms for nonparametric solution to the minimum probability of error classification problem," *IEEE Trans. Comput.*, vol. C-27, pp. 648–659, 1978.
[8] R. Durbin and D. Willshaw, "An analogue approach to the traveling salesman problem using an elastic net method," *Nature*, vol. 326, pp. 689–691, 1987.
[9] F. Girosi and T. Poggio, "Networks and the best approximation property," *Biolog. Cybern.*, vol. 63, no. 3, pp. 169–176, 1990.
[10] D. Harrison and D. L. Rubinfeld, "Hedonic prices and the demand for clean air," *J. Environ. Econom. Manag.* vol. 5, pp. 81–102, 1978.
[11] J. B. Hampshire and A. H. Waibel, "A novel objective function for improved phoneme recognition using time-delay neural networks," *IEEE Trans. Neural Networks*, vol. 1, pp. 216–228, 1990.
[12] E. J. Hartman, J. D. Keeler, and J. M. Kowalski, "Layered neural networks with Gaussian hidden units as universal approximations," *Neural Comput.*, vol. 2, no. 2, pp. 210–215, 1990.
[13] Y. H. Hu, S. Palreddy, and W. J. Tompkins, "Customized ECG beat classifier using mixture of experts," *Neural Networks Signal Process.*, pp. 459–464, 1995.
[14] J.-H. Hwang, S.-R. Lay, M. Maechler, R. D. Martin, and J. Schimert, "Regression modeling in back-propagation and projection pursuit learning," *IEEE Trans. Neural Networks*, vol. 5, pp. 342–353, May 1994.
[15] R. A. Jacobs and M. I. Jordan, "Learning piecewise control strategies in a modular neural network architecture," *IEEE Trans. Syst., Man Cybern.*, vol. 23, pp. 337–345, 1993.
[16] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton, "Adaptive mixtures of local experts," *Neural Comput.*, vol. 3, no. 1, pp. 79–87, 1991.
[17] M. I. Jordan and R. A. Jacobs, "Hierarchical mixtures of experts and the EM algorithm," *Neural Comput.*, vol. 6, no. 2, pp. 181–214, 1994.
[18] B. H. Juang and S. Katagiri, "Discriminative learning for minimum error classification," *IEEE Trans. Signal Processing*, vol. 40, pp. 3043–3054, 1992.
[19] B. H. Juang, R. J. Perdue, Jr., and D. L. Thomson, "Deployable automatic speech recognition systems: Advances and challenges," *AT&T Tech. J.*, vol. 74, pp. 45–56, 1995.
[20] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Sci.*, vol. 220, pp. 671–680, 1983.
[21] T. Kohonen, G. Barna, and R. Chrisley, "Statistical pattern recognition with neural networks: Benchmarking studies," in *Proc. IEEE ICNN*, 1988, vol. 1.
[22] S. Lee and J. C.-J. Pan, "Unconstrained handwritten numeral recognition based on radial basis competitive and co-operative networks with spatio-temporal feature representation," *IEEE Trans. Neural Networks*, vol. 7, pp. 455–474, Mar. 1996.
[23] J. A. Leonard and M. A. Kramer, "Radial basis function networks for classifying process faults," *IEEE Contr. Syst. Mag.*, vol. 11, pp. 31–38, Apr. 1991.
[24] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Commun.*, vol. COMM-28, pp. 84–95, 1980.
[25] G. C. McDonald and R. C. Schwing, "Instabilities of regression estimates relating air pollution to mortality," *Technometr.*, vol. 15, pp. 463–482, 1973.
[26] G. J. McLachlan and K. E. Basford, *Mixture Models: Inference and Application to Clustering.* New York: Marcel Dekker, 1988.
[27] D. Miller, A. Rao, K. Rose, and A. Gersho, "A global optimization method for statistical classifier design," *IEEE Trans. Signal Processing*, vol. 44, pp. 3108–3122, Dec. 1996.
[28] J. Moody and C. J. Darken, "Fast learning in networks of locally-tuned processing units," *Neural Comput.*, vol. 1, no. 2, pp. 281–294, 1989.
[29] N. Morgan and H. A. Boulard, "Neural networks for statistical recognition of continuous speech," *Proc. IEEE*, vol. 83, pp. 742–772, 1995.
[30] B. Mulgrew, "Applying radial basis functions," *IEEE Signal Processing Mag.*, vol. 13, pp. 50–65, Mar. 1996.
[31] M. Niranjan and V. Kadirkamanathan, "A nonlinear model for time series prediction and signal interpolation," in *Proc. ICASSP*, 1991, vol. 3, pp. 1713–1716.
[32] M. J. D. Powell, "Radial basis functions for multivariable interpolation: A review," in *Algorithms for Approximation.* Oxford, U.K.: Clarendon, 1987, pp. 143–167.
[33] A. Rao, D. Miller, K. Rose, and A. Gersho, "An annealing approach to parsimonious modeling in statistical regression," submitted for publication.
[34] K. Rose, E. Gurewitz, and G. C. Fox, "Vector quantization by deterministic annealing," *IEEE Trans. Inform. Theory*, vol. 38, pp. 1249–1258, 1992.
[35] ———, "Constrained clustering as an optimization method," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 15, pp. 785–794, 1993.
[36] ———, "Statistical mechanics and phase transitions in clustering," *Phys. Rev. Lett.*, vol. 65, pp. 945–948, 1990.
[37] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, *Parallel Distributed Processing.* Cambridge, MA: MIT Press, 1986.
[38] D. M. Titterington, A. F. M. Smith, and U. E. Makov, *Analysis of Finite Mixture Distributions.* New York: Wiley, 1985.
[39] S. R. Waterhouse and A. J. Robinson, "Non-linear prediction of acoustic vectors using hierarchical mixtures of experts," *Neural Inform. Process. Syst.*, vol. 7, pp. 835–842, 1995.
[40] A. S. Weigend, M. Mangeas, and A. N. Srivastava, "Nonlinear gated experts for time series: Discovering regimes and avoiding overfitting," *Int. J. Neural Syst.*, vol. 4, pp. 373–399, 1995.
[41] L. Xu and M. I. Jordan, "On convergence properties of the EM algorithm for Gaussian mixtures," *Neural Comput.*, vol. 8, pp. 129–151, 1996.

**Ajit V. Rao** (S'93) was born in Bangalore, India, in 1971. He received the B.Tech. degree in electronics and communication engineering in 1992 from the Indian Institute of Technology, Madras, and the M.S. degree in electrical and computer engineering in 1993 from the University of California, Santa Barbara, where he is currently pursuing the Ph.D degree.

In the summer of 1995, he worked as an intern in the Speech Coding Group, Texas Instruments Inc., Dallas, TX. His current interests are speech and image coding and statistical pattern recognition.

**David Miller** (S'87–M'95) received the B.S.E. degree from Princeton University, Princeton, NJ, in 1987, the M.S.E. degree from the University of Pennsylvania, Philadelphia, in 1990, and the Ph.D. degree from the University of California, Santa Barbara in 1995, all in electrical engineering.

From January 1988 through January 1990, he was employed by General Atronics Corporation, Wyndmoor, PA. Since August 1995, he has been an Assistant Professor of Electrical Engineering at the Pennsylvania State University, University Park. His research interests include source and channel coding, image compression, statistical pattern recognition, and neural networks.

Dr. Miller received the National Science Foundation Career Award in 1996 for the continuation of his research on learning algorithms for neural networks.

**Kenneth Rose** (S'85–M'91) received the B.Sc. (summa cum laude) and M.Sc (magna cum laude) degrees in electrical engineering from Tel-Aviv University, Tel-Aviv, Israel, in 1983 and 1987, respectively, and the Ph.D. degree in electrical engineering from the California Institute of Technology (Caltech), Pasadena, in 1990.

From July 1983 to July 1988, he was employed by Tadiran Ltd., Israel, where he carried out research in the areas of image coding, transmission through noisy channels, and general image processing. From September 1988 to December 1990, he was a graduate student at Caltech. In January 1991, he joined the Department of Electrical and Computer Engineering, University of California, Santa Barbara, where he is currently an Associate Professor. His research interests are in information theory, source and channel coding, pattern recognition, image coding and processing, and nonconvex optimization in general.

Dr. Rose was co-recipient of the William R. Bennett Prize Paper Award of the IEEE Communications Society in 1990.

**Allen Gersho** (S'58–M'64–SM'78–F'81) received the B.S. degree from the Massachusetts Institute of Technology, Cambridge, in 1960 and the Ph.D. degree from Cornell University, Ithaca, NY, in 1963.

He was at Bell Laboratories from 1963 to 1980 and is currently Professor of Electrical and Computer Engineering at the University of California, Santa Barbara (UCSB). His current research activities are in signal compression methodologies and algorithm development for speech, audio, image, and video coding. He holds patents on speech coding, quantization, adaptive equalization, digital filtering, and modulation and coding for voiceband data modems. He is co-author with R. M. Gray of the book *Vector Quantization and Signal Compression* (Boston, MA: Kluwer, 1992) and co-editor of two books on speech coding.

Dr. Gersho served as a member of the Board of Governors of the IEEE Communications Society from 1982 to 1985 and is a member of various IEEE technical, award, and conference management committees. He has served as Editor of *IEEE Communications Magazine* and Associate Editor of the IEEE TRANSACTIONS ON COMMUNICATIONS. He received NASA "Tech Brief" awards for technical innovation in 1987, 1988, and 1992. In 1980, he was co-recipient of the Guillemin–Cauer Prize Paper Award from the Circuits and Systems Society. He received the Donald McClennan Meritorious Service Award from the IEEE Communications Society in 1983, and in 1984, he was awarded an IEEE Centennial Medal. In 1992, he was co-recipient of the 1992 Video Technology Transactions Best Paper Award from the IEEE Circuits and Systems Society.