DETERMINISTICALLY ANNEALED MIXTURE OF EXPERTS MODELS FOR STATISTICAL REGRESSION

Ajit Rao\(^1\), David Miller\(^2\), Kenneth Rose\(^1\), Allen Gersho\(^3\)

\(^1\)Dept. of Electrical and Computer Engineering, University of California, Santa Barbara, CA 93106.
\(^2\)Dept. of Electrical Engineering, Penn. State University, University Park, PA 16802.

ABSTRACT
A new and effective design method is presented for statistical regression functions that belong to the class of mixture models. The class includes the hierarchical mixture of experts (HME) and the normalized radial basis functions (NRBF). Design algorithms based on the maximum likelihood (ML) approach, which emphasize a probabilistic description of the model, have attracted much interest in HME and NRBF models. However, their design objective is mismatched to the original squared-error regression cost and the algorithms are easily trapped by poor local minima on the cost surface. In this paper, we propose an extension of the deterministic annealing (DA) method for the design of mixture-based regression models. We construct a probabilistic framework, but unlike the ML method, we directly optimize the squared-error regression cost, while avoiding poor local minima. Experimental results show that the DA method outperforms standard design methods for both HME and NRBF regression models.

1. MIXTURE OF EXPERTS REGRESSION

In recent years, there has been growing interest in learning methods for regression functions that can be statistically interpreted as mixture models or mixture of experts (ME) models. The ME regression function takes the form:

\[
g(x) = \sum_j P[j|x] f(x, \Lambda_j),
\]

where \(P[j|x]\) is a non-negative weight of association between input, \(x\) and the \(j\)th “local expert regression function”, \(f(x, \Lambda_j)\). Each local expert, \(f(x, \Lambda_j)\), is usually a constant, linear or simple nonlinear function of \(x\) and depends on the parameter set, \(\Lambda_j\).

The weights of association can be naturally interpreted as a probability distribution since \(\sum_j P[j|x] = 1\).

Hence, the ME model can be interpreted as a probabilistic partition of the input space - every point in the input space belongs in probability to partition cells, each of which is governed by a local regression model.

Some popular neural network approaches to regression, such as the hierarchical mixture of experts (HME) \cite{4} and normalized radial basis functions (NRBF) \cite{8}, can be formulated as mixture of experts models.

1.1. The learning problem

Consider the problem of learning a regression function from a “training set”, \(T = \{(x_i, y_i)\}, i = 1, 2, \ldots, N\). The natural choice of learning criterion is the minimization of the average squared-error cost measured over the training set,

\[
\min_{\{\Lambda_j\},\{P[j|x]\}} D = \frac{1}{N} \sum_i \|y_i - g(x_i)\|^2
\]

However in \cite{3} and \cite{4}, a maximum likelihood (ML) criterion,

\[
\max_{\{\Lambda_j\},\{P[j|x]\}} L = \sum_i \log \sum_j P[j|x_i] e^{-\|y_i - f(x_i, \Lambda_j)\|^2},
\]

was preferred for a few reasons - Firstly ML training is fast, and performed better than minimum squared-error training in experiments. Further, ML training can realized by the Expectation Maximization (EM) algorithm \cite{2} which has useful convergence properties. Also, ML training yields “competitive” solutions in which
relatively few experts are strongly activated for any
given input; Squared-error training yields “co-operative”
solutions, where many experts typically contribute to
give an output. In [3], competitive models were favored
based on the advantages of a localized representation.

However, despite the advantages and promising re-
results of the ML algorithm, we believe that the minimum
squared-error cost is a more appropriate training crite-
ron. Specifically, we note that the ML criterion of (3)
encourages an individual fit between output \( y \), and each
expert \( f(x_i, A_j) \), rather than the co-operative fit based
on the ME output \( g(x_i) \). While gradient ascent on the
ML cost surface sometimes minimizes the squared-error
better than direct gradient descent on the squared-error
cost, all that this really suggests is that the squared-
error surface is more complex than the ML surface,
with numerous poor local optima to trap simple descent
methods. Rather than abandon the squared-error crite-
nion to avoid the design difficulty, we suggest a more
powerful method, deterministic annealing (DA), for the
minimization.

2. DETERMINISTIC ANNEALING

Our work is based on the DA method proposed in the
and its extensions to incorporate structurally constrained
data clustering problems [5][6] which finds practical use
in the design of statistical classifiers [7] and General-
ized Vector Quantizers [9] for source-coding applications. The DA method is based on the interpretation of
a mixture model as a randomized space partition. The
degree of randomization of this partition can be mea-
sured by the Shannon entropy,

\[
H = -\frac{1}{N} \sum_{i=1}^{N} \sum_{j} P[j|x_i] \log P[j|x_i]. \tag{4}
\]

We first pose the problem of optimizing the regres-
sion cost, \( D \) of equation (2), while constraining the
Shannon entropy, \( H = H_0 \). This constrained optimiza-
tion problem may be written as the minimization of the
corresponding Lagrangian,

\[
\min_{\{P_j[j|x_i]\}, \{A_j\}} \left( D - TH \right) \tag{5}
\]

where the Lagrange parameter, \( T \) is referred to as the
“temperature” to emphasize a compelling analogy to
statistical physics. Equation (5) reminds us of the defi-
nition of the Helmholtz free energy of a thermodynamic
system, where \( D \) is the thermodynamic energy of a
physical system, \( T \) is the temperature and \( H \) is the
entropy. The temperature (Lagrange multiplier) de-
termines a balance of energy (cost) and entropy (ran-
domness). Minimizing the Lagrangian, \( D - TH \), we
minimize the Helmholtz free energy, and in fact, seek
isothermal equilibrium at the given temperature, \( T \).
Of particular importance is the case of \( T \to 0 \) which cor-
responds to direct minimization of \( D \), our ultimate ob-
tective. This suggests the possibility of implementing
an annealing process, that is, gradually lowering the
temperature while maintaining the system at thermal
equilibrium. Such a process allows one to avoid many of
the local minima of the energy \( D \). Since this method
is not a stochastic method like simulated annealing,
but instead based on the optimization of the determin-
istically computed expectation of the Helmholtz free
energy, it is considerably faster.

We initialize the algorithm with a very high value
of \( T \). At this temperature, we must maximize the en-
tropy of associating inputs with regions. The solution
chooses all the probability distributions to be uniform
and all the local models, \( \{A_j\} \) to be equal to a single
global model of the data. Effectively, a single region
would suffice to represent the entire data. As the tem-
perature is gradually lowered, in steps, optimization is
carried out at each temperature to choose the param-
ters of the probability distribution and the local model
parameters, \( \{A_j\} \) that minimize the Lagrangian. As
\( T \to 0 \), the Lagrangian reduces to the regression cost,
\( D \).

2.1. The NRBF structure

In the NRBF architecture,

\[
P_j(x) \propto e^{-\frac{(x-m_j)^2}{2\sigma^2}} \tag{6}
\]
defines the association probability which depends on
the relative closeness to each of the “prototypes”, \( \{m_j\} \).
Furthermore, \( f(x, A_j) = A_j \), i.e. we make use of con-
stant local models. To design an NRBF regression
function from the training data, one must optimize the
locations of the prototypes, \( \{m_j\} \) and the local models,
\( \{A_j\} \) to minimize the regression cost, \( D \). A common
design approach[8] adopted for the design of NRBF-based
regression functions can be summarized in two-steps :

- Find the prototypes, \( \{m_j\} \) that minimizes a clus-
tering (VQ-like) cost in the \( X \) space.
- With this initialization for \( \{m_j\} \), use a gradient
descent algorithm on the regression cost, \( D \), of
  equation (2) to optimize \( \{m_j\}, \{A_j\} \).

While this algorithm is quick, a significant problem
with it is that despite the heuristically reasonable ini-
tialization that the first step offers to the second (cost
minimization) step, the method can be easily trapped in poor local minima on the complex cost surface. We will demonstrate this problem with some examples in the next section and show that the DA approach effectively overcomes this shortcoming to generate better regression functions.

2.2. The HME structure

The hierarchical mixture of experts (HME) regression function is organized as a tree. The leaves of the tree represent simple local regression models (experts) which are weighted and combined, as they “traverse” to the root node, where the final regression estimate, \( g(x) \) is computed. As an example, for a simple two-level, binary-tree HME architecture \(^1\), the output

\[
g(x) = \sum_j g_j(x) \sum_k g_{kj}(x)f(x, \Lambda_{jk}),
\]

where \( g_j(x) \propto e^{y_j x} \) and \( g_{kj}(x) \propto e^{y_{jk} x} \) define the probability distributions. This architecture can be viewed as a mixture model, where each local model, \( f(x, \Lambda_{jk}) \) is selected based on a probability that is the (tree-structured) product of the probabilities, \( g_{kj}(x) \) and \( g_j(x) \).

Jordan and Jacobs\(^4\) suggested the maximization of the likelihood function (3), as an effective method to design HME regression functions. However, although this method does sometimes minimize the regression cost better than gradient descent on the squared-error cost function, its design objective is mismatched to the original regression cost. In the results section, we will demonstrate that our novel DA method significantly outperforms both likelihood maximization, and gradient descent on the squared-error cost.

3. RESULTS

We applied our DA-based design method to the HME and NRBFF architectures and compared the average squared-error performance with those obtained by conventional design methods for each architecture. In this section, we demonstrate that the DA method clearly outperforms the Two-step (2ST) method for the NRBFF architecture and outperforms both the maximum likelihood (ML) and gradient descent (GD) methods for the HME architecture. The experiments are performed for different values of \( K \), the number of local models used for regression. Note that in the case of NRBFF regression functions, \( K \) is the number of prototypes and in the case of binary HME trees with \( l \) levels, \( K = 2^l \).

Since, for both HME and NRBFF structures, the performance of the competing methods depends on the initialization used, we attempted to remove the bias introduced due to poor initialization by allowing each competing method to use ten different initializations with only the best result obtained among those runs compared with the result obtained by DA. Since the regression function obtained by DA is generally independent of the initialization, a single DA run will suffice. Further, in the case of HME regression functions, we use linear local models.

Our experiments are performed over three benchmark examples from the StatLib dataset archive. \(^2\)

- **The Boston home value prediction problem** : The goal is to use a training set of data from 506 homes in the Boston area to predict the median price of each home from 13 features which are believed to influence it. Since the features have different dynamic ranges, we first normalize each feature to unit variance before designing regression functions for them. Results are shown in tables 1 (a) and (b).

- **Prediction of mortality rate** : We consider the prediction of the age-adjusted mortality rate in a locality from 15 factors that may have possibly influenced it. Since we have data on only 60 localities, we used the entire dataset for training. Results are shown in tables 2 (a) and (b).

- **Estimation of fat content of meat** : The fat content of meat can be measured by techniques of analytical chemistry, but it is a slow and time-consuming process. In this experiment, we used a dataset of quick absorption measurements \(^3\) and the corresponding fat content as determined by analytical chemistry to learn to predict the fat content from the measurements. The data consists of a training set of size 173 and a test set of size 43. The results using NRBFF and HME regression functions for the prediction are shown in tables 3 (a) and (b).

Clearly, for all three examples, the deterministic annealing method outperforms the standard design methods for both mixture model architectures. Note that, in table 3(b), allowing the ML approach to use a larger network size does not necessarily improve the performance over the test set, although performance over the training set improves marginally.

\(^{1}\)Note that our method is not restricted to binary or two-level trees. This example is given only for ease of understanding.


\(^{3}\)The Tecator Infratec Food and Feed Analyzer measures the absorption of electro-magnetic waves in 100 different frequency bands.
4. ACKNOWLEDGMENTS

The authors wish to thank Professors Michael I. Jordan and Prof. Robert A. Jacobs for providing their HME design software.

5. REFERENCES


Table 1: Comparison of average squared-error for the Boston home data problem using (a) NRBF architecture and (b) the binary HME tree. $K$ is the corresponding network size.

<table>
<thead>
<tr>
<th>K</th>
<th>DA</th>
<th>2-Step</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>87.7</td>
<td>87.7</td>
</tr>
<tr>
<td>2</td>
<td>19.7</td>
<td>23.78</td>
</tr>
<tr>
<td>4</td>
<td>12.9</td>
<td>19.34</td>
</tr>
<tr>
<td>6</td>
<td>12.6</td>
<td>13.74</td>
</tr>
<tr>
<td>10</td>
<td>6.5</td>
<td>15.72</td>
</tr>
</tbody>
</table>

Table 2: Comparison of average squared-error for the environmental data problem using the (a) NRBF architecture and (b) the binary-HME tree. $K$ is the corresponding network size.

<table>
<thead>
<tr>
<th>K</th>
<th>DA (tr)</th>
<th>DA (te)</th>
<th>2-Step (tr)</th>
<th>2-Step (te)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>159.89</td>
<td>168.25</td>
<td>159.89</td>
<td>168.25</td>
</tr>
<tr>
<td>2</td>
<td>52.9</td>
<td>58.8</td>
<td>131.43</td>
<td>159.68</td>
</tr>
<tr>
<td>4</td>
<td>28.6</td>
<td>32.9</td>
<td>119.82</td>
<td>137.99</td>
</tr>
<tr>
<td>6</td>
<td>27.3</td>
<td>40.1</td>
<td>74.89</td>
<td>83.73</td>
</tr>
</tbody>
</table>

Table 3: Comparison of average squared-error for fat-content estimation using (a) the NRBF architecture and (b) the binary HME tree. $K$ is the number of basis functions. 'tr' and 'te' refer to training and test sets respectively.

<table>
<thead>
<tr>
<th>K</th>
<th>DA (tr)</th>
<th>DA (te)</th>
<th>GD (tr)</th>
<th>GD (te)</th>
<th>ML (tr)</th>
<th>ML (te)</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>8.3</td>
<td>11.5</td>
<td>14.1</td>
<td>18.1</td>
<td>15.1</td>
<td>23.9</td>
</tr>
<tr>
<td>8</td>
<td>6.9</td>
<td>9.8</td>
<td>12.8</td>
<td>17.2</td>
<td>12.5</td>
<td>39.7</td>
</tr>
</tbody>
</table>