# Classification Using Localized Mixtures of Experts

Perry Moerland
IDIAP, CP 592, Martigny, Switzerland
E-mail: Perry.Moerland@idiap.ch

**Abstract**. A mixture of experts consists of a gating network that learns to partition the input space and of experts networks attributed to these different regions. This paper focuses on the choice of the gating network. First, a *localized* gating network based on a mixture of linear latent variable models is proposed that extends a gating network introduced by Xu *et al.* [9], based on Gaussian mixture models. It is shown that this localized mixture of experts model, can be trained with the Expectation Maximization algorithm. The localized model is compared on a set of classification problems, with mixtures of experts having single or multi-layer perceptrons as gating network. It is found that the standard mixture of experts with feed-forward networks as gate often outperforms the other models.

## 1 Introduction

A *mixture of experts* [5] is a probabilistic model that can be interpreted as a mixture model for estimating conditional probability distributions. The model consists of a *gating* network that divides the problem into smaller problems and makes *expert* networks specialize on each of these subproblems. In terms of a mixture model, the expert networks correspond to conditional component densities and the gating network to input-dependent mixture coefficients. This interpretation of mixtures of experts as mixture models enables training them with the Expectation Maximization (EM) algorithm [5]. Note that, the gating network splits the data in a *soft* way, allowing several experts to be selected at a time.

Since the gating network deals with the decomposition in smaller tasks, the choice of the type of gating network is an important one. The standard mixture of experts model has a single-layer perceptron with a soft-max activation function as gate [5]. This leads to a division of the input space by soft hyper-planes with decision boundaries that are simply connected and convex. An alternative approach is the use of what Weigend *et al.* coined *gated experts* [8]. In this model, the gate is a multi-layer perceptron (MLP) with a soft-max output activation function. This enables far more complex decompositions with non-linear decision boundaries. A third approach is to divide the input space with soft hyper-ellipsoids using normalized Gaussian kernels [9], each localized to a specific expert. Finally, also a hierarchical mixture of experts [5] has been proposed which has a tree structure. The leaves of the tree contain the expert networks and the non-terminal nodes contain the gating networks. This model also enables complex decompositions while using simple gating networks.

We introduce an extension of Xu's gating network based on Gaussian mixture models (GMMs) [9]. This extension uses mixtures of linear *latent variable* models [2] as a gating network. This choice is motivated by the fact that mixtures of latent variable models can be interpreted as a mixture of *constrained* Gaussians, that offer a more flexible alternative for GMMs. Since this type of gate decomposes the input space with soft hyper-ellipsoids, we will refer to the whole model as a *localized* mixture of experts. The standard mixture of experts model and our extension to a localized mixture of experts with a mixture of linear latent variable models as a gate, are described in section 2. It is also outlined how the localized model can be trained by the EM algorithm.

The experimental evaluation of the localized mixtures of experts and other mixture of expert models on a set of classification problems, is described in section 3. The goal of these experiments was to evaluate the influence of the choice of the gating network on the overall performance of the whole model. The gating networks evaluated are: mixtures of latent variable models, GMMs, single-layer perceptrons, and MLPs.

A more elaborate version of this paper can be found in [6].

## 2 Mixtures of Experts

In section 2.1, GMMs and mixtures of linear latent variable models are briefly described. Then, the basics of the mixture of experts model are recalled. Finally, it is outlined how the mixture models of section 2.1 can be used as a gating network leading to a localized mixture of experts. It is also sketched how this model can be trained via the EM algorithm, but the reader is referred to [6] for a more detailed description.

### 2.1 Mixture Models

A mixture model is defined as a linear combination of $m$ component densities $p_j(\mathbf{x})$:

$$p(\mathbf{x}) = \sum_{j=1}^{m} \alpha_j p_j(\mathbf{x}), \tag{1}$$

where the $\alpha_j$ are the mixing coefficients which are non-negative and sum to one. A standard tool for density estimation is a GMM where the component distributions are Gaussian with a covariance matrix $\mathbf{\Sigma}_j$ that is chosen to be full, diagonal or spherical: $p_j(\mathbf{x}) \sim \mathcal{N}(\boldsymbol{\mu}_j, \mathbf{\Sigma}_j)$. The parameters of a GMM can be determined in a maximum likelihood framework by the EM algorithm [1]. A disadvantage of GMMs is that they either impose strong constraints on the covariance matrices (spherical or diagonal) or no constraints at all (full).

A more flexible alternative for GMMs are the recently introduced mixtures of latent variable models [2]. A latent variable model relates a $l$-dimensional latent vector $\mathbf{z}$ to a $d$-dimensional ($l < d$) observed data vector $\mathbf{x}$ by defining a noise model and a prior on the distribution of the latent variables:

$$\mathbf{x} = \mathbf{W}\mathbf{z} + \boldsymbol{\mu} + \varepsilon. \tag{2}$$

The prior distribution of the latent variables is a simple Gaussian distribution $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ over the latent space. The first two terms on the right-hand side of (2) are the mean $\boldsymbol{\mu}$, and the $(d \times l)$ generative matrix $\mathbf{W}$. that maps the latent space into the data space. The result is convolved in data space with a Gaussian distribution $\varepsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$ with a restricted covariance matrix $\mathbf{R}$. With $\mathbf{R} = \sigma^2\mathbf{I}$, the latent variable model is called probabilistic principal component analysis [2]; with $\mathbf{R} \sim$ diagonal matrix, the latent variable model is standard factor analysis [2]. The advantage of such linear latent variable models is that the distribution of the observed data vectors is also Gaussian: $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{R} + \mathbf{W}\mathbf{W}^T)$. This means in specific, that the model can be viewed as a flexible (through the choice of $l$) way of capturing the covariance structure $\mathbf{R} + \mathbf{W}\mathbf{W}^T$ of the $d$-dimensional observed data using less parameters ($l + dl$) than if one would model the full covariance matrix in the observed data space ($d(d+1)/2$ parameters). The resulting mixture model (1) is a linear combination of linear latent variable component distributions:

$$p_j(\mathbf{x}) \sim \mathcal{N}(\boldsymbol{\mu}_j, \mathbf{R}_j + \mathbf{W}_j\mathbf{W}_j^T). \tag{3}$$

With $\mathbf{R}_j$ isotropic, the model is called a mixture of principal component analysers (MPCA) [2] and with $\mathbf{R}_j$ diagonal, a mixture of factor analysers (MFA) [4]. The parameters of a mixture of linear latent variable models can be estimated by the EM algorithm [2, 4].

These mixtures of latent variable models have been applied successfully to density estimation problems [6], where it is shown that they are a more flexible alternative for GMMs and often lead to better scores in terms of likelihood.

### 2.2 Localized Models

A mixture of experts consists of $m$ experts, the outputs $\mathbf{y}_i(\mathbf{x})$ of which are weighted by the outputs of a gating network $g_i(\mathbf{x})$ for input vector $\mathbf{x}$:

$$\mathbf{y}(\mathbf{x}) = \sum_{j=1}^{m} g_j(\mathbf{x})\mathbf{y}_j(\mathbf{x}).$$

A probabilistic interpretation of a mixture of experts can be given in the context of mixture models for conditional probability distributions (with a soft-max activation function for the gating network to have non-negative outputs that sum to one):

$$p(\mathbf{t}|\mathbf{x}) = \sum_{j=1}^{m} g_j(\mathbf{x})\phi_j(\mathbf{t}|\mathbf{x}), \tag{4}$$

where the $\phi_j$ represent the conditional densities of target vector $\mathbf{t}$ for expert $j$. This interpretation makes that a mixture of experts can be trained in a maximum likelihood framework by the EM algorithm, that

| Data set | # attr. | # classes | # examples | # attr. (after pre-processing) | missing data |
|---|---|---|---|---|---|
| Dermatology | 34 | 6 | 366 | 34 | • |
| Glass | 9 | 6 | 214 | 9 | |
| Letter | 16 | 26 | 20,000 | 16 | |
| NIST | 256 | 10 | 20,000 | 256 | |
| Optical | 64 | 10 | 3,823 | 64 | |
| Pen | 16 | 10 | 7,494 | 16 | |
| Soybean | 35 | 19 | 683 | 134 | • |
| Vowel | 10 | 11 | 990 | 10 | |
| Waveform | 21 | 3 | 600 | 21 | |
| Waveform-noise | 40 | 3 | 600 | 40 | |

Table 1: Properties of the data sets used in the experiments.

decouples the learning of the expert and gating networks. The M-step of the EM algorithm for a gating network (single or multi-layer) with a soft-max output function results in a non-linear optimization problem which requires iterative techniques [5].

A method for reducing the M-step for the gating network to a one-pass calculation has been proposed in [9]: a gating network consisting of normalized kernels each localized to a specific expert (by applying Bayes' rule):

$$g_j(\mathbf{x}) = P(j|\mathbf{x}) = \frac{\alpha_j p_j(\mathbf{x})}{\sum_i \alpha_i p_i(\mathbf{x})}, \qquad (5)$$

where $\sum_i \alpha_i = 1$, $\alpha_i \geq 0$, and the $p_i$'s are probability density functions; thus the gating network outputs $g_j$ sum to one and are non-negative. The numerator in eq. (5) can be interpreted as the component of a simple mixture model (1).

This choice of the gating network leads to the following probability model for the entire mixture of experts model (substituting (5) in (4)):

$$p(\mathbf{t}|\mathbf{x}) = \sum_{j=1}^{m} \frac{\alpha_j p_j(\mathbf{x})}{\sum_i \alpha_i p_i(\mathbf{x})} \phi_j(\mathbf{t}|\mathbf{x}). \qquad (6)$$

To obtain a one-pass solution for the gating network parameters, maximum likelihood estimation is not performed on this conditional density, but on the joint density [9]:

$$p(\mathbf{x}, \mathbf{t}) = p(\mathbf{t}|\mathbf{x}) p(\mathbf{x}) = \sum_{j=1}^{m} \alpha_j p_j(\mathbf{x}) \phi_j(\mathbf{t}|\mathbf{x}),$$

which by maximum likelihood leads to the following error function on the training data $\{\mathbf{x}^n, \mathbf{t}^n\}$:

$$E = -\sum_n \ln \sum_{j=1}^{m} \alpha_j p_j(\mathbf{x}^n) \phi_j(\mathbf{t}^n|\mathbf{x}^n).$$

The basic idea of the EM algorithm is that the minimization of this error function can be simplified if each pattern could be associated with exactly one expert (indicated by so-called missing variables $z_j^n$ equal to one for only one expert and zero for the others). This is done by iteratively repeating a two step procedure (consisting of an E-step and a M-step). The E-step consists of calculating the expected values of the missing variables:

$$\mathcal{E}(z_j^n) = \frac{\alpha_j p_j(\mathbf{x}^n) \phi_j(\mathbf{t}^n|\mathbf{x}^n)}{\sum_{i=1}^{m} \alpha_i p_i(\mathbf{x}^n) \phi_i(\mathbf{t}^n|\mathbf{x}^n)} = h_j(\mathbf{x}^n, \mathbf{t}^n).$$

In the M-step, the so-called expected complete error function is minimized (or decreased, for *generalized* EM) with respect to the parameters of the expert networks and the gate. The error function can be interpreted as the sum of an unsupervised part that encourages good density estimation (gate) and a supervised part that encourages correct classification (experts). The expert error function and consequently the M-step for the expert networks is identical to the one obtained in [5] for standard mixtures of experts. We focus therefore on the gating error function: $-\sum_n \sum_{j=1}^m h_j(\mathbf{x}^n, \mathbf{t}^n) \ln(\alpha_j p_j(\mathbf{x}^n))$. This is almost the error function that is minimized in the M-step when applying the EM algorithm to a simple mixture model (see, for example [1]). The only difference is in the definition of their posteriors $h$ that in the case of a localized mixture of experts, include both input and output values and thus incorporate the supervised errors at the output of the expert networks.

Since we have not yet defined the probability densities $p_j(\mathbf{x})$ in (6), any mixture model that can be trained with EM could be used as a gating network in this framework. This makes it possible to use not only

| Gate | test | 5×2cv |
|---|---|---|
| Spherical | 79.7(3.28) | |
| Diagonal | 79.7(2.73) | |
| MFA-2 | 80.5(2.00) | |
| MFA-1 | 80.6(2.85) | |
| MPCA-3 | 80.8(2.16) | |
| Full | 81.2(1.40) | |
| MFA-3 | 81.6(1.39) | |
| MPCA-2 | 81.8(1.34) | |
| MLP | 82.2(1.63) | |
| MPCA-1 | 82.3(1.28) | |
| Perceptron | 83.2(1.16) | < |

Table 2: Classification results on the waveform data with a mixture of 3 experts. Scores are in percentages of correct classification.

| Gate | test | 5×2cv |
|---|---|---|
| Full | 76.8(1.38) | |
| Spherical | 77.4(1.50) | |
| MFA-1 | 77.5(1.08) | |
| Diagonal | 77.6(1.74) | |
| MFA-2 | 77.6(0.92) | |
| MPCA-2 | 78.0(1.66) | |
| MPCA-1 | 78.1(1.57) | |
| MFA-3 | 78.2(1.59) | |
| MPCA-3 | 78.7(0.77) | |
| MLP | 79.6(3.44) | |
| Perceptron | 81.7(1.02) | < |

Table 3: Classification results on the waveform-noise data with a mixture of 3 experts. Scores are in percentages of correct classification.

GMMs as in [9] but also mixtures of latent variable models such as MPCAs and MFAs (3). These alternative choices for the localized mixture of experts have been evaluated in the experiments described in the next section.

## 3 Experiments

In the literature, experiments with localized mixtures of experts based on GMMs have mainly been performed on isolated problems. The paper by Xu *et al.* [9] in which the localized model has been proposed, reports only the results on a toy regression problem. Gated experts have not yet been used often and only on some isolated problems. Weigend *et al.* applied gated experts to several time series problems especially ones with different regimes [8]. A gated expert model was used on a problem in automatic speech recognition in [7]. A thorough experimental evaluation of standard mixture of experts has been done by Steve Waterhouse [7] in the DELVE framework. For classification problems, however, DELVE suffers from a lack of data sets. We, therefore, chose for a different experimental set-up to evaluate our localized mixtures of experts and the influence of the choice of the gating network in general.

### 3.1 Experimental Set-Up

The experiments with the mixtures of experts were performed on a range of classification problems out of the Irvine repository and part of the NIST special database 3 of handwritten digits (Table 1). The desired outputs are based on the 1-of-$c$ coding scheme with one output for each class. The raw data has been pre-processed in various ways and the reader is referred to [6] for a more detailed description of the preprocessing.

The expert networks are single-layer perceptrons with a soft-max output function, except for the gated experts where the experts were chosen to have the same MLP architecture as the gate.

Training of the localized mixtures of experts consisted of two phases. In the first phase, the mixture model for the gating network was trained in an unsupervised fashion with $k$-means and the EM algorithm to find a good initial configuration (see [6] for a more detailed description). In the second phase, the whole model was trained in the EM framework. The M-step for each of the experts consisted of three iterations of the scaled conjugate gradient algorithm [1]. The M-step for the gate is as described in section 2, based on the M-step for the corresponding mixture model. For the experiments with the standard mixtures of experts and the gated experts, the M-step for each of the experts and the gating network again consisted of three iterations of the scaled conjugate gradient algorithm.

The $5 \times 2$cv test (a paired $t$-test) [3] for testing the statistically significant difference, was used on all data sets except the NIST data. In the $5 \times 2$cv test, five replications of twofold cross-validation are performed. On the NIST data, only one run has been performed and in this case, McNemar's test was used to test for statistical significance [3]. We used a fixed training (15025 examples written by 140 persons) and test (4975 examples written by 48 persons not in the training set) set.

The entries in the tables with results are the average percentage of correctly classified test patterns over 10 simulations (except for the NIST data); the standard deviation is given between parentheses. A $<$-sign in the tables with results, indicates whether the score on the test set is significantly better (80%) than the one on the previous row. MFA-$l$ and MPCA-$l$ denote a mixture of latent variable models with a dimension of latent space equal to $l$. Full, diagonal, and spherical, refer to the type of covariance matrix used in the GMM-based gate.

## 3.2 Artificial Data

As a first test, experiments were performed on two often used artificial classification problems with code for generating the data at the Irvine repository: the waveform and the waveform-noise data (the last two rows of Table 1).

The results on the waveform and waveform-noise data are in tables 2 and 3. The gated experts, indicated by MLP, had 4 hidden units in both the expert and gating networks. On both waveform benchmarks, the standard mixture of experts significantly outperforms the alternative models. The results for the other models do not permit a further ordering.

To get some insight in the solution found by a localized mixture of 3 experts, Figure 1 shows the projection of the waveform data on its two leading principal components. Each of the 3 classes turns out to lie on the edge of a triangle. Also shown are the three Gaussians of the MFA-1 model that was used as a gating network. The three clusters lie close to the vertices of the triangle and the subproblem solved by each of the experts is therefore effectively reduced to a two-class problem for separating the two edges out of this vertex. This might also explain why the results are better with a perceptron gate. In the latter case, the gating network is namely far less localized and performs a sort of averaging (which is known to often improve accuracy) of the perceptron experts. Since the accuracy of a single perceptron on the waveform data (see [6]) is already satisfactory, the decomposition found by the localized mixture of experts does not improve the results.

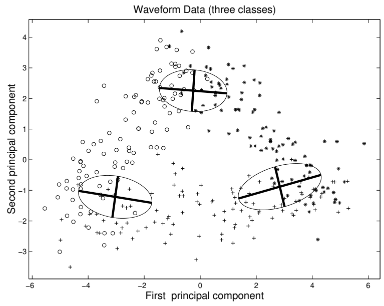


Figure 1: Projection of the waveform data on the 2 leading principal components with the ellipses indicating the components of the MFA-1 gate trained on this data.

## 3.3 Real-World Data

Do the good results with a standard mixture of experts on both artificial data sets, carry over to real-world data? To answer this question, experiments have been performed on the other databases listed in Table 1. The results are shown in Table 4. The method with the highest score and the ones that are not significantly worse (80% with the $5 \times 2$cv test or McNemar's test) are set in bold face. If most of the methods performed equally well, the ones that are significantly worse are set in italics. For the experiments with a MPCA/MFA gate many different dimensions of the latent space were tried and the ones giving the best results are shown here (more details in [6]). This means that the results as presented here for the MPCA/MFA gate are favorably biased. For the gated experts, the number of hidden units was either 4 (for dermatology and glass) or 10 (for the others).

While all choices for the gating network lead to good results on at least one benchmark, the best results are clearly obtained with the standard mixture of experts and the gated experts. Variability of the results is also often larger for localized models than for the standard mixture of experts.

When comparing the various localized mixtures of experts, it is clear that the results are not uniform. None of the GMMs or mixtures of latent variable models can be preferred over the others. More specifically, there seems to be no correlation between the performance of the gate as a density estimator in the input space and the classification results.

A criterion for quantifying the difference

| Data set | MFA | MPCA | spherical | diagonal | full | perceptron | MLP |
|---|---|---|---|---|---|---|---|
| Dermatology (2) | 96.3(0.8) | 96.3(0.9) | **96.6(1.0)** | 94.9(1.4) | 94.7(1.0) | 96.3(0.9) | **95.5(1.3)** |
| Dermatology (4) | 96.0(1.0) | **96.0(0.7)** | 95.8(1.5) | 93.0(2.0) | 92.0(2.0) | **96.4(0.7)** | **95.0(1.8)** |
| Glass (2) | 63.7(2.9) | *63.7(2.8)* | 64.1(2.6) | 63.0(3.4) | 63.4(3.7) | 64.7(2.1) | 63.8(2.5) |
| Glass (4) | 64.1(1.8) | 64.5(2.1) | 65.8(2.3) | *64.0(2.3)* | *60.2(3.4)* | 66.2(1.9) | 62.9(3.5) |
| Letter (2) | **81.6(0.6)** | 80.4(0.9) | 79.8(0.2) | 79.2(0.3) | 81.5(0.8) | 81.4(1.3) | **82.3(0.8)** |
| Letter (10) | 87.5(1.0) | **88.3(1.3)** | 86.0(0.9) | 85.3(2.3) | **89.1(1.3)** | **90.1(0.6)** | 87.5(0.7) |
| NIST (2) | 93.6 | 94.2 | 94.5 | **94.7** | 92.7 | 94.2 | **95.1** |
| NIST (4) | 93.6 | 95.3 | 94.9 | 94.8 | 91.5 | **96.1** | 95.1 |
| Optical (4) | 95.7(0.4) | 96.0(0.5) | 95.9(0.3) | 96.1(0.7) | 95.8(0.5) | **96.7(0.4)** | **96.2(0.6)** |
| Optical (10) | 95.0(0.7) | 95.6(1.1) | 95.3(0.6) | 95.5(0.8) | 95.9(0.4) | **97.0(0.4)** | 96.7(0.3) |
| Pen (4) | 98.4(0.2) | 98.1(0.8) | 97.2(2.8) | *97.7(0.6)* | 97.4(2.1) | 98.7(0.3) | 98.6(0.2) |
| Pen (10) | 95.5(3.0) | *96.9(0.8)* | 97.7(0.7) | 95.6(1.7) | 96.2(2.0) | 98.9(0.2) | 98.8(0.2) |
| Soybean (2) | 90.4(1.2) | 89.4(1.5) | 89.9(2.7) | 90.0(2.9) | 89.3(1.8) | 90.1(1.5) | 89.7(2.4) |
| Soybean (4) | 83.6(12.0) | 87.6(6.0) | 88.8(5.7) | 79.5(13.6) | 83.3(8.5) | 90.5(1.6) | 90.1(1.6) |
| Vowel (4) | 80.2(2.6) | 80.7(3.4) | **79.3(3.3)** | **80.9(2.6)** | 78.0(3.2) | 77.1(3.2) | **79.9(4.2)** |
| Vowel (11) | 83.2(2.1) | 83.1(2.8) | 81.8(2.8) | 82.9(2.2) | *80.3(4.4)* | 82.4(2.4) | 83.7(2.7) |

Table 4: Results of the experiments with a mixture of experts and different gating networks. Scores are in percentages of correct classification. The best scores are set in bold and the worst scores in italics. The number of experts is indicated between parentheses after the name of each data set.

between feed-forward gates and localized gates, is the entropy of the gating outputs: $-\sum_n \sum_{j=1}^{m} g_j(\mathbf{x}^n) \ln g_j(\mathbf{x}^n)$. On most of the data sets described in this paper, the entropy of the feed-forward gates was greater than the entropy of the localized gates by one order of magnitude. This illustrates that standard mixtures of experts are far less localized and attribute patterns across the experts, if this happens to reduce the total error. This seems to confirm Jordan and Jacobs' claim that the soft splits of the standard mixture of experts reduce the variance of the model [5] which might explain the better results obtained using feed-forward gates.

## 4 Conclusions

Mixtures of latent variable models can be used as a gating network in a localized mixtures of experts model trained via the EM algorithm. However, a comparison of these localized models on 10 data sets for classification, shows that they are often outperformed by standard mixtures of experts and gated experts. This might be explained by the *softer* splits obtained when using feed-forward gates which decreases the variance of the model.

Note that, the comparison between the standard mixture of experts and the gated experts is not completely fair, since for the latter model the experts are MLPs. The trade-off between more complex experts and the number of experts is a subject for further research.

**Acknowledgements**
The author gratefully acknowledges the Swiss National Science Foundation (FN:21-45621.95) for their support of this research.

## References

[1] C. M. Bishop. *Neural Networks for Pattern Recognition.* Oxford University Press, Oxford, 1995.

[2] C. M. Bishop and M. E. Tipping. A hierarchical latent variable model for data visualization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3):281–293, March 1998.

[3] T. G. Dietterich. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, 10(7):1895–1923, 1998.

[4] Z. Ghahramani and G. E. Hinton. The EM algorithm for mixtures of factor analyzers. Technical Report CRG-TR-96-1, University of Toronto, 1996.

[5] M. I. Jordan and R. A. Jacobs. Hierarchical mixtures of experts and the EM algorithm. *Neural Computation*, 6(2):181–214, 1994.

[6] P. Moerland. Localized mixtures of experts. IDIAP-RR 98-14, IDIAP, `http://www.idiap.ch/`, 1998.

[7] S. R. Waterhouse. *Classification and regression using mixtures of experts.* PhD thesis, Cambridge University Engineering Department, October 1997.

[8] A. S. Weigend, M. Mangeas, and A. N. Srivastava. Nonlinear gated experts for time series: Discovering regimes and avoiding overfitting. *International Journal of Neural Systems*, 6:373–399, 1995.

[9] L. Xu, M. I. Jordan, and G. E. Hinton. An alternative model for mixtures of experts. In G. Tesauro, D. S. Touretzky, and T. K. Leen, editors, *Advances in NIPS*, volume 7, pages 633–640, Cambridge MA, 1995. MIT Press.