

DEPARTMENT OF STATISTICS
The Pennsylvania State University
University Park, PA 16802 U.S.A.

TECHNICAL REPORTS AND PREPRINTS

Number 09-03: April 2009

**Mixture of Regressions with Predictor-dependent
Mixing Proportions**

Derek S. Young and David R. Hunter

Mixtures of Regressions with Predictor-Dependent Mixing Proportions

Derek S. Young and David R. Hunter
Penn State Department of Statistics Technical Report 09-03

April 28, 2009

Abstract

We extend the standard mixture of linear regressions model by allowing mixing proportions to be modeled nonparametrically as a function of the predictors. This framework allows for more flexibility in the modeling of the mixing proportions than the fully parametric mixture of experts model, which we also discuss. We present an EM-like algorithm for estimation of the new model. We also provide simulations demonstrating that our nonparametric approach can provide a better fit than the parametric approach in some instances and can serve to validate and thus reinforce the parametric approach in others. We also analyze and interpret two real data sets using the new method.

Keywords: EM algorithms, hierarchical mixture of experts, mixture models, mixtures of regressions

1 Introduction

In a typical multivariate finite mixture model, the m -dimensional vectors $\mathbf{Y}_1, \dots, \mathbf{Y}_n$ are a simple random sample from an m -component mixture distribution such that \mathbf{Y}_i has density

$$f(\mathbf{y}_i; \boldsymbol{\psi}) = \sum_{j=1}^m \lambda_j g(\mathbf{y}_i; \boldsymbol{\theta}_j), \quad (1)$$

where $m > 1$ is fixed (and assumed known for now) and the λ_j , called the weights (or *mixing proportions*) for the components, are positive and sum to unity. The density g is assumed to come from a parametric family with parameter $\boldsymbol{\theta}_j \in \Theta_j \subseteq \mathbb{R}^q$ where Θ_j is open in \mathbb{R}^q . The mixture density f is parameterized

by $\boldsymbol{\psi} \in \boldsymbol{\Psi}$, where $\boldsymbol{\Psi}$ represents the parameter space for all unknown parameters in the mixture model, i.e., $(\lambda_1, \dots, \lambda_m, \boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_m)$.

Suppose now a vector of predictors, say $\mathbf{X}_i = (X_{i,1}, \dots, X_{i,p})^\top$ for $p < n$, is also observed with each response \mathbf{Y}_i . The goal is to describe the conditional distribution of $\mathbf{Y}_i | \mathbf{X}_i$ using a mixture of linear regressions with assumed Gaussian errors. Thus, Equation (1) becomes

$$f(\mathbf{y}_i; \mathbf{x}_i, \boldsymbol{\psi}) = \sum_{j=1}^m \lambda_j \phi(\mathbf{y}_i; \mathbf{x}_i^\top \boldsymbol{\beta}_j, \sigma_j^2), \quad (2)$$

where $\phi(\cdot; \mathbf{x}_i^\top \boldsymbol{\beta}_j, \sigma_j^2)$ is the normal probability density function with mean $\mathbf{x}_i^\top \boldsymbol{\beta}_j$ and variance σ_j^2 for some $(\boldsymbol{\beta}_j, \sigma_j^2) \in \mathbb{R}^p \times \mathbb{R}_*^+$.

The uses of mixtures of regressions fall into two primary categories. The first involves estimating a set of regression coefficients for all observations coming from a possibly unknown number of heterogeneous classes. This scenario arises when it seems inaccurate to assume that a single regression adequately explains the relationship between the variables at hand. An example of this scenario is depicted in Figure 1(a), which shows data from an experiment on the perception of musical tones (Cohen, 1980). Mixtures of regressions have been extensively studied in the econometrics literature and were first introduced by Quandt (1972) as the *switching regimes* (or *switching regressions*) problem. A switching regimes system is often compared to *structural change* in a system (Quandt and Ramsey, 1978). A structural change assumes the system depends deterministically on some observable variables (such as a time variable), but switching regimes implies one is unaware of what causes the switch between regimes. In the case where it is assumed there are two heterogeneous classes, Quandt (1972) characterized the switching regimes problem “by assuming that nature chooses between regimes with probabilities λ and $1 - \lambda$.” Quandt and Ramsey (1978) also developed a moment-generating function for parameter estimation.

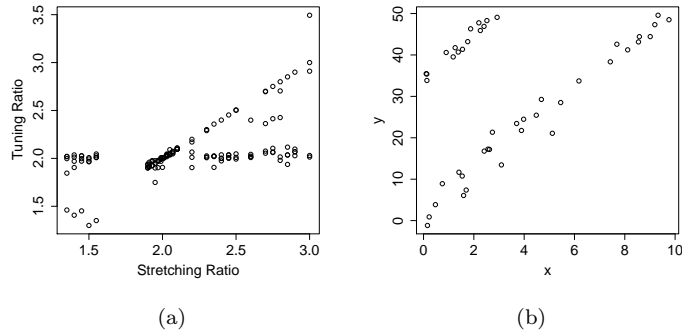


Figure 1: Plot (a) is a scatterplot of a data set from a tone perception study reported in Cohen (1980). Plot (b) shows simulated data illustrating masked outliers.

A second use for mixtures of regressions is in outlier detection or robust regression estimation. This occurs under the assumption that one regression plane can adequately model the data, but there is an apparent class heterogeneity because of large variances attributed to some observations, which are considered outliers. Viele and Tong (2002) consider this approach when discussing *masked outliers*, which they describe as outliers that “cannot be detected individually by standard techniques.” Pena, Rodríguez, and Tiao (2003) used a split and recombine (SAR) procedure to identify possible clusters in a sample, since masked outliers often appear as clusters. Figure 1(b) is a simulated data set with masked outliers, similar to those analyzed in Viele and Tong (2002) and Pena *et al.* (2003). The cluster of points in the upper left of the scatterplot are high leverage outliers which traditional methods fail to detect (see Justel and Pena, 1996).

Given the m -component mixture of regressions model (2) and a new observation $(\mathbf{y}_{n+1}, \mathbf{x}_{n+1})$, one might ask which of the m regression functions in the model should be used to predict the value of the response \mathbf{y}_{n+1} . According to the model, each regression should occur with probability equal to its corre-

sponding λ_j , but this might not be realistic if \mathbf{x}_{n+1} contains some information about the relative weights. To reflect this possibility in the notation, we may replace model (2) by

$$f(\mathbf{y}_i; \mathbf{x}_i, \boldsymbol{\psi}) = \sum_{j=1}^m \lambda_j(\mathbf{x}_i) \phi(\mathbf{y}_i; \mathbf{x}_i^T \boldsymbol{\beta}_j, \sigma_j^2). \quad (3)$$

How should we model $\lambda_j(\mathbf{x}_i)$ in equation (3)? One way is to assume a parametric form, such as a logistic function, which introduces new parameters requiring estimation. This is the idea of the *hierarchical mixtures of experts* (HME) procedure (Jacobs, Jordan, Nowlan, and Hinton, 1991), which is commonly used in neural networks. The HME procedure is a variant on tree-based methods – a context somewhat different from the one we describe. Since a parametric form of $\lambda_j(\mathbf{x}_i)$ may be too restrictive, we propose in this article a nonparametric estimate of $\lambda_j(\mathbf{x}_i)$ that uses ideas from kernel density estimation.

The remainder of the article discusses the HME model, then introduces our more flexible nonparametric method for modeling the mixing weights as functions of the predictors. We also provide an EM-like algorithm for estimation using our method. Finally, we illustrate the method using both simulated data sets and real data sets. EM algorithms for both the traditional mixture-of-regressions model and the HME model are included as appendices. The code for the estimation procedures is included in a package `mixtools` (Young, Benaglia, Chauveau, Hunter, Elmore, Xuan, Hettmansperger, and Thomas, 2008) for the R statistical computing environment (R Development Core Team, 2008).

2 Hierarchical Mixtures of Experts

The hierarchical mixtures of experts (HME) model comes from the *statistical learning* (or *machine learning*) literature [see Jordan and Jacobs (1992), Jordan and Jacobs (1994), and Jordan and Xu (1995) for discussion]. In a statistical learning problem, the goal is to categorize each of a set of individuals into one

of m classes based on some measurements. In the *supervised learning* problem, some of the individuals (called the training data) are already categorized, and the goal is to build a *learner*, which is a model used to predict the outcome for new subjects. In the *unsupervised learning* problem, the goal is to cluster unlabeled training data by partitioning a set of features into a number of statistical classes. Mixture models generally can be considered an unsupervised learning method, even though classification of individuals is not always the goal of mixture modeling. Although we compare our proposed (mixture model-based) method to HME in this article, as Hastie, Tibshirani, and Friedman (2001) points out, the HME procedure is really a variant of tree-based methods such as CART and MARS, so the focus of HME models is often on supervised learning. We will return to this point after formally defining the HME model.

The architecture of the HME model involves a set of functions (the leaves in the tree structure of Figure 2) called *expert networks* that are combined together by a classifier (the non-leaf nodes) called *gating networks*. The gating networks split the feature space into regions where one expert network seems more appropriate than the other. For illustration, we present a HME model with two levels in Figure 2.

The top gating network in a two-level HME has the output

$$\lambda_j(\mathbf{x}_i, \boldsymbol{\tau}) = \frac{\exp\{\mathbf{x}_i^T \boldsymbol{\tau}_j\}}{\sum_{r=1}^{m_1} \exp\{\mathbf{x}_i^T \boldsymbol{\tau}_r\}}, \quad (4)$$

where $j = 1, \dots, m_1$, $\boldsymbol{\tau}_j \in \mathbb{R}^p$ is an unknown gating parameter vector and $\boldsymbol{\tau} = (\boldsymbol{\tau}_1^T, \dots, \boldsymbol{\tau}_{m_1}^T)^T$. Since the $\lambda_j(\mathbf{x}_i, \boldsymbol{\tau})$ of Equation (4) are nonnegative and sum to one, they produce a probabilistic split (or *soft split*) that partitions the feature space into m_1 regions. We now have another level of gating networks, where each of the m_1 gating networks has its own output, given by

$$\lambda_{j,l}(\mathbf{x}_i, \boldsymbol{\omega}_j) = \frac{\exp\{\mathbf{x}_i^T \boldsymbol{\omega}_{j,l}\}}{\sum_{r=1}^{m_2} \exp\{\mathbf{x}_i^T \boldsymbol{\omega}_{j,r}\}}, \quad (5)$$

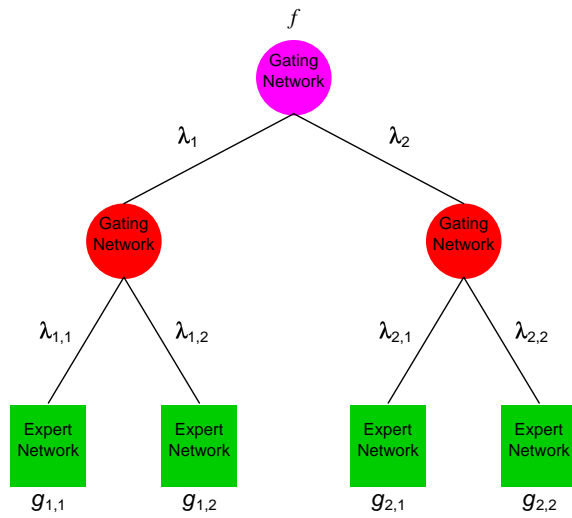


Figure 2: A diagram of a HME model with two levels. The density f is approximated by the weighted sum of lower gating networks, which are the weighted sum of expert networks: $f = \sum_{j=1}^2 \lambda_j \sum_{l=1}^2 \lambda_{j,l} g_{j,l}$. For economy of notation, we write $g(y_i; \mathbf{x}_i, \boldsymbol{\theta}_{j,l})$ as $g_{j,l}$, $\lambda_j(\mathbf{x}_i, \boldsymbol{\tau})$ as λ_j , and so on.

where $l = 1, \dots, m_2$ and $\boldsymbol{\omega}_j = (\boldsymbol{\omega}_{j,1}^T, \dots, \boldsymbol{\omega}_{j,m_2}^T)^T$. This means we have probabilistically split each of the previous m_1 regions into its own set of m_2 subregions. These subregions correspond to the expert networks, which model the output variable, y_i , according to the probability density function $g(y_i; \mathbf{x}_i, \boldsymbol{\theta}_{j,l})$ where $\boldsymbol{\theta}_{j,l}$ is a component-specific parameter vector. These probability densities typically belong to an exponential family; for simplicity, we assume the densities are normal. Combining all of the preceding yields the mixture density

$$f(y_i; \mathbf{x}_i, \boldsymbol{\psi}) = \sum_{j=1}^{m_1} \lambda_j(\mathbf{x}_i, \boldsymbol{\tau}) \sum_{l=1}^{m_2} \lambda_{j,l}(\mathbf{x}_i, \boldsymbol{\omega}_j) g(y_i; \mathbf{x}_i, \boldsymbol{\theta}_{j,l}), \quad (6)$$

where $\boldsymbol{\psi}$ is the parameter vector consisting of all component-specific parameters and all gating parameter vectors for each gating network. Thus, we have a mixture model where the mixing proportions are determined by the gating network models.

We have presented only a two-level HME model in (6) and Figure 2 depicts the simple case in which $m_1 = m_2 = 2$. The model can be extended to more than two levels of gating networks (Jordan and Xu, 1995), but the notation becomes rather cumbersome. Also, when there is only one gating network, there is no hierarchical structure and the model is simply a *mixture of experts* (ME). Models similar to the ME model appear elsewhere. In the psychology literature, Nagin (1999) implements such a model to statistically link group membership to predictors when analyzing developmental trajectories. Also, Yau, Lee, and Ng (2003) propose

$$\lambda_j(\mathbf{x}_i) = \frac{\exp\{\mathbf{x}_i^T \boldsymbol{\tau}_j + \delta_i\}}{\sum_{r=1}^{m_1} \exp\{\mathbf{x}_i^T \boldsymbol{\tau}_r + \delta_i\}}, \quad (7)$$

where δ_i represents an unobservable random effect due to the i^{th} subject affecting the value of the j^{th} mixing proportion.

As Jordan and Xu (1995) point out, the mixtures of linear regressions model can be viewed as representing one end of a continuum and tree-based methods (like CART and MARS) the other end, while the HME model “interpolates

smoothly between these extremes.” At one end of this continuum, the tree-based methods rigidly partition the covariate space, with each point in this space belonging to exactly one of the m subgroups. At the other end, the plain mixture of regressions model (2) assigns each subgroup a fixed probability that is independent of the covariates. In between these methods, the HME model has gating networks that allow the probability of subgroup membership to depend on the covariate vector. Since this is a property shared by the method we propose here, our method may be viewed as another such interpolation method.

Bayesian methods have been developed for the HME architecture (Jacobs, Peng, and Tanner, 1997). However, we present in Appendix B an EM algorithm for maximum likelihood estimation. The output from this algorithm will later be compared with estimates using our nonparametric method.

3 Nonparametric Predictor-Dependent Mixing Proportions

It is not difficult to imagine situations in which the logistic curves of an HME model are not appropriate for modeling the dependence of the mixing proportions λ_j on the predictors \mathbf{x} . Consider the simulated data set of Figure 3(a), which comes from a mixture of simple linear regressions.

In this simulation, independent realizations x_1, \dots, x_{500} of the predictor are generated from a uniform distribution on $(0, 100)$. Then, the component — one or two — is chosen at random, with component 1 having probability $[1 + \cos(x/10)]/2$. Finally, realizations of the response, y_1, \dots, y_{500} , are generated as

$$Y_i \sim \begin{cases} N(x_i, 6), & \text{if component 1;} \\ N(-50 + 2x_i, 7), & \text{if component 2.} \end{cases} \quad (8)$$

After applying the standard mixture-of-regressions EM algorithm given in Appendix A for model (2) in which the λ_j are assumed constant, we automatically

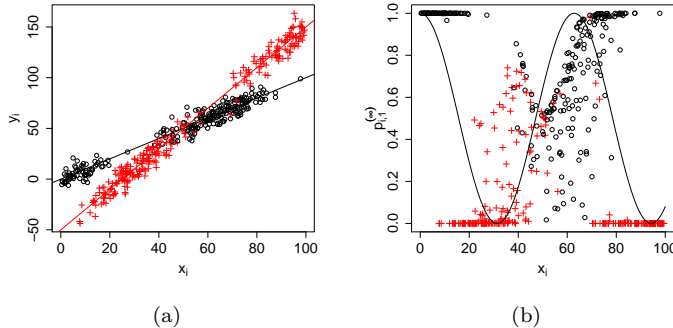


Figure 3: (a) Scatterplot of a simulated data set. Note that the colors and shapes, which distinguish between the two subpopulations and are known only because these data are simulated, are assumed to be unobserved. (b) Plot of the posterior membership probabilities $p_{i,1}^{(\infty)}$ from an EM algorithm versus the predictors. The curve represents the way the true component labels were generated. In both plots, black o's denote component 1 and red +'s denote component 2.

obtain the quantities $p_{i,1}^{(\infty)}$ of equation (30), where

$$p_{i,1}^{(\infty)} = \frac{\lambda_1^{(\infty)} f_1^{(\infty)}(y_i - x_i \beta_1^{(\infty)})}{\lambda_1^{(\infty)} f_1^{(\infty)}(y_i - x_i \beta_1^{(\infty)}) + \lambda_2^{(\infty)} f_2^{(\infty)}(y_i - x_i \beta_2^{(\infty)})}$$

and $f_j^{(\infty)}$ is the normal density with mean zero and standard deviation $\sigma_j^{(\infty)}$. These $p_{i,1}^{(\infty)}$ may be called the *posterior probabilities* of membership in component 1 for the i^{th} observation (this term is sensible if $\lambda_j^{(\infty)}$ is considered the prior probability of membership, and thus it gives a convenient way to refer to the $p_{i,j}^{(\infty)}$). Plotting the posterior probabilities against the predictor values results in Figure 3(b). The solid curve in Figure 3(b) is the curve $[1 + \cos(x/10)]/2$ by which the component probabilities were generated originally.

The parametric form of $\lambda_j(x)$ in the HME model does not have the flexibility to capture the oscillating shape of the true relationship shown in Figure 3(b). As an alternative, we propose a nonparametric approach, similar to kernel regression, for modeling the mixing proportions.

When we obtain estimates for In estimating λ_j using an EM algorithm, what we are actually doing is finding the marginal expectation of the indicator variables

$$Z_{i,j} = \mathbf{I}\{\text{observation } i \text{ belongs to component } j\},$$

$1 \leq j \leq m$. Our nonparametric approach attempts to utilize instead a locally weighted average using a kernel density. As in nonparametric regression (of $Z_{i,j}$ on \mathbf{X}_i), we wish to estimate the expression

$$\lambda_j(\mathbf{x}_i) = \mathbf{E}[Z_{i,j} | \mathbf{X}_i = \mathbf{x}_i] \quad (9)$$

of model (3). Exploiting well-studied ideas of kernel regression (Nadaraya, 1964; Watson, 1964), we propose to approximate this conditional probability by

$$\lambda_j(\mathbf{x}_i) = \frac{\sum_{l=1}^n Z_{l,j} \mathcal{K}_{\mathbf{h}}(\mathbf{x}_i - \mathbf{x}_l) \mathbf{I}(\mathbf{A}_l)}{\sum_{l=1}^n \mathcal{K}_{\mathbf{h}}(\mathbf{x}_i - \mathbf{x}_l) \mathbf{I}(\mathbf{A}_l)}, \quad (10)$$

where

$$\mathcal{K}_{\mathbf{h}}(\mathbf{x}_i - \mathbf{x}_l) = \frac{1}{h_1 \cdots h_p} \mathcal{K}\left(\frac{x_{i,1} - x_{l,1}}{h_1}, \dots, \frac{x_{i,p} - x_{l,p}}{h_p}\right) \quad (11)$$

and

$$\mathbf{A}_l = \left\{ \left| \frac{x_{i,m} - x_{l,m}}{h_m} \right| \leq 1 \quad \forall m = 1, \dots, p \right\}.$$

Here, \mathcal{K} denotes a (multivariate) kernel density function operating on p arguments where p is the length of the predictor vector and $\mathbf{h} = (h_1, \dots, h_p)^T$ is a vector of bandwidths. For simplicity, we deal strictly with the univariate case ($p = 1$) in our discussion.

There are several issues concerning kernel-based nonparametric methods, such as choice of bandwidth and selection of kernel type. We briefly outline the methods we use, but a deeper treatment of these and other issues in kernel-based nonparametric methods may be found in Wand and Jones (1995).

We use cross-validation (CV) for bandwidth selection in our simulations, which is a data-driven method. Denote the full data set by S , which we partition

into a training set $S - S_j$ and test set S_j , $j = 1, 2, \dots, N$, where N corresponds to the number of times we partition the data. In other words, S_j and $S - S_j$ will always be disjoint, but the S_j themselves need not be disjoint nor cover S . For each time we partition the data, the intent is to build a learner based on the training data, which we then evaluate at the test set. We denote the actual value of the predictor-dependent mixing proportions by $\lambda(x_i)$ and build the learner $\lambda_{S_j}(x_i)$ based on the j^{th} training set.^[9] Then, we estimate $\lambda_{S_j}(x_i)$ evaluated at the values in the j^{th} test set S_j . The CV criterion we use has the form

$$CV(h) = \sum_{j=1}^N \sum_{i \in S_j} [\lambda(x_i) - \lambda_{S_j}(x_i)]^2 \quad (12)$$

and we select the bandwidth that minimizes this criterion. For our simulations, we restrict attention to a range of integer values for a rough approximation, but actually h can be any positive real number.

Some commonly used kernels are the Gaussian kernel

$$\mathcal{K}(x) = \frac{1}{\sqrt{2\pi}} \exp\{-x^2/2\} \quad (13)$$

and those belonging to the symmetric beta family

$$\mathcal{K}(x) = \frac{1}{\text{Beta}(1/2, \gamma + 1)} (1 - x^2)_+^\gamma, \quad (14)$$

where $\gamma = 0, 1, 2, \dots$. In the symmetric beta family, specific kernels include uniform, Epanechnikov, biweight, and triweight for $\gamma = 0, 1, 2$, and 3, respectively.

Other options include the cosine kernel

$$\mathcal{K}(x) = \frac{1}{2}(1 + \cos(\pi x)) \quad (15)$$

and the optcosine kernel

$$\mathcal{K}(x) = \frac{\pi}{4}(\cos(\pi x/2)). \quad (16)$$

⁹Notice that we have suppressed the subscript denoting component membership as our simulations come from a 2-component mixture and hence have only one mixing proportion.

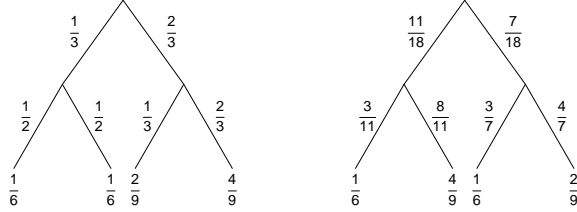


Figure 4: Two possible trees for the same set of mixing proportions.

Another important difference between our local model and the HME model is that we cannot impose a hierarchical structure for the local setting. Consider the tree in Figure 2. In the HME setting, each mixing proportion is the product of inverse logit (denoted by inv.logit) functions. For instance, the mixing proportion for $g(y_i; \mathbf{x}_i, \boldsymbol{\theta}_{1,1})$ is

$$\begin{aligned} \lambda_1(\mathbf{x}_i, \boldsymbol{\tau})\lambda_{1,1}(\mathbf{x}_i, \boldsymbol{\omega}_1) &= \text{inv.logit}(\mathbf{x}_i^T \boldsymbol{\tau})\text{inv.logit}(\mathbf{x}_i^T \boldsymbol{\omega}_1) \\ &= \frac{\exp\{\mathbf{x}_i^T (\boldsymbol{\tau} + \boldsymbol{\omega}_1)\}}{1 + \exp\{\mathbf{x}_i^T \boldsymbol{\tau}\} + \exp\{\mathbf{x}_i^T \boldsymbol{\omega}_1\} + \exp\{\mathbf{x}_i^T (\boldsymbol{\tau} + \boldsymbol{\omega}_1)\}}, \end{aligned}$$

so we have the same functional form for each i . However, in the nonparametric case, suppose we have the mixing proportions $\{1/6, 1/6, 2/9, 4/9\}$ as in Figure 4. Then there are multiple trees that can result in this set of mixing proportions. Therefore, our nonparametric approach makes the parameters of a hierarchical structure nonidentifiable. All comparisons and simulations in the remainder of this chapter will be made with the mixture of experts (ME) model and not the HME model.

Simulations presented later show that our method does better than the ME approach according to a CV criterion and the mean square error (MSE) when there is curvature or oscillatory behavior in the form of $\lambda_j(\mathbf{x}_i)$. In cases where ME solutions are close to the nonparametric solutions, the latter are still valu-

able because they provide verification of the ME model assumptions. Furthermore, we obtain an increase in the observed log likelihood using our new model compared to the fits (utilizing an EM algorithm) obtained from both the ME procedure and the model with predictor-free mixing proportions. How to measure the significance of the increase has not yet been investigated and we return to a brief discussion regarding this topic in the conclusion. Before proceeding with the simulations and examples, we briefly define an iterative algorithm used for estimation.

3.1 IGLE Algorithm

In the EM algorithm presented in Appendix A, estimation of the mixing proportions in the maximization step (M-Step) is straightforward. The estimate for λ_j is just the mean of the posterior membership probabilities for a given j . However, this is not true when modeling the mixing proportions as a function of the predictors. In the ME setting, an iteratively reweighted least squares (IRLS) loop is required for estimation. In the local setting, we still use an “EM-like” algorithm, although we have no analogue of the ascent property of a true EM algorithm. Since we are still interested in global (i.e., not dependent on \mathbf{x}) values for all other parameters in the model, we iterate between estimating global values and then estimating local mixing proportions. Thus, we call our algorithm an iterative global/local estimation (IGLE) algorithm.

Consider a partition of the parameter vector $\boldsymbol{\psi}$, say $(\boldsymbol{\psi}_1^T, \boldsymbol{\psi}_2^T)^T$, where $\boldsymbol{\psi}_2^T$ consists of the predictor-dependent mixing proportions at each \mathbf{x}_i and $\boldsymbol{\psi}_1^T$ consists of all the remaining parameters. We assume a value for the bandwidth, h , is provided. We now define the IGLE algorithm.

IGLE Algorithm

1. For a given $\boldsymbol{\psi}^{(t)}$ at the t^{th} iteration, $t = 0, 1, \dots$, estimate $\boldsymbol{\psi}_1$ globally via one step of an EM algorithm. Call this estimate $\boldsymbol{\psi}_1^{(t+1)}$.

2. Conditioned on $\boldsymbol{\psi}_1^{(t+1)}$, estimate

$$\lambda_j(\mathbf{x}_i)^{(t+1)} = \frac{\sum_{l=1}^n p_{l,j}^{(t+1/2)} \mathcal{K}_{\mathbf{h}}(\mathbf{x}_i - \mathbf{x}_l)}{\sum_{l=1}^n \mathcal{K}_{\mathbf{h}}(\mathbf{x}_i - \mathbf{x}_l)}, \quad (17)$$

where

$$\begin{aligned} p_{l,j}^{(t+1/2)} &= \frac{\lambda_j(\mathbf{x}_l)^{(t)} (2\pi\sigma_j^{2(t+1)})^{-1/2} \exp\left\{-\frac{1}{2\sigma_j^{2(t+1)}} \left(y_l - \mathbf{x}_l^T \boldsymbol{\beta}_j^{(t+1)}\right)^2\right\}}{\sum_{k=1}^m \lambda_k(\mathbf{x}_l)^{(t)} (2\pi\sigma_k^{2(t+1)})^{-1/2} \exp\left\{-\frac{1}{2\sigma_k^{2(t+1)}} \left(y_l - \mathbf{x}_l^T \boldsymbol{\beta}_k^{(t+1)}\right)^2\right\}} \\ &= \left[1 + \sum_{k \neq j}^m \frac{\lambda_k(\mathbf{x}_l)^{(t)} \sigma_j^{(t+1)}}{\lambda_j(\mathbf{x}_l)^{(t)} \sigma_k^{(t+1)}} \exp\left\{\frac{1}{2} \left[\frac{1}{\sigma_j^{2(t+1)}} \left(y_l - \mathbf{x}_l^T \boldsymbol{\beta}_j^{(t+1)}\right)^2\right.\right.\right. \\ &\quad \left.\left.\left. - \frac{1}{\sigma_k^{2(t+1)}} \left(y_l - \mathbf{x}_l^T \boldsymbol{\beta}_k^{(t+1)}\right)^2\right]\right\}\right]^{-1}. \end{aligned}$$

The $p_{l,j}^{(t+1/2)}$ values are the posterior membership probabilities obtained from the EM algorithm implemented in the global estimation step. Thus, we have the estimate $\boldsymbol{\psi}_2^{(t+1)}$.

3. Iterate until a stopping criterion is attained.

We say IGLE is an ‘‘EM-like’’ algorithm because the global step does perform estimation via an EM algorithm, which is conditioned on the mixing proportions. However, in the local step there is no maximization of a global or local likelihood function. The estimates of the mixing proportions are simply replaced with locally weighted estimates. This differs from, say, estimating all of the parameters locally in a nonparametric mixture of regressions model or estimating the mixing proportions globally and the remaining parameters locally in a semiparametric mixture of regressions model (see Huang (2006)). The estimation procedures for both of these models do maximize a local likelihood.

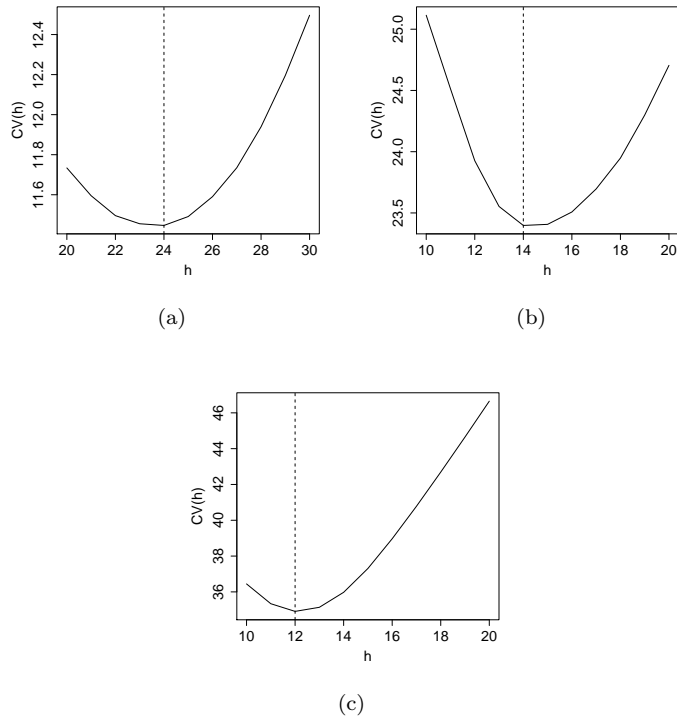


Figure 5: Plots of the cross-validation criterion $CV(h)$ versus the bandwidth (h) for sample sizes (a) 250, (b) 500, and (c) 1000 and $\lambda(x_i)$ as given in (20).

A log likelihood local to \mathbf{x}_0 is defined as

$$\begin{aligned} \ell(\Psi(\mathbf{x}_0)) = \sum_{i=1}^n \log \left[\mathcal{K}_h(\mathbf{x}_i - \mathbf{x}_0) \sum_{j=1}^m \lambda_j (2\pi\sigma_j^2(\mathbf{x}_0))^{-1/2} \right. \\ \left. \times \exp \left\{ -\frac{1}{2\sigma_j^2(\mathbf{x}_0)} \left(y_i - \mathbf{x}_i^\top (\boldsymbol{\beta}_j(\mathbf{x}_0)) \right)^2 \right\} \right]. \quad (18) \end{aligned}$$

Notice the flexibility gained by modeling the σ_j^2 's and $\boldsymbol{\beta}_j$'s as functions of \mathbf{x}_0 .

4 Examples

4.1 Simulations

Two underlying functional forms are considered for the values of the $\lambda(x_i)$'s in the following simulations. First, we simulate the realizations x_1, \dots, x_n from

	250	500	1000
EM	0.0071	0.0068	0.0066
ME EM	0.0087	0.0078	0.0072
IGLE	0.0023	0.0014	0.0008

Table 1: Average mean squared errors for 1000 data sets for each $n \in \{250, 500, 1000\}$ using model (19).

the random variable $X_i \sim Unif(0, 100)$ and the realizations y_1, \dots, y_n from the random variable

$$Y_i \sim \begin{cases} N(x_i, 6), & \text{with probability } \lambda(x_i); \\ N(-50 + 2x_i, 7), & \text{with probability } 1 - \lambda(x_i), \end{cases} \quad (19)$$

where

$$\lambda(x_i) = 1 - \left(\frac{x_i - 50}{100} \right)^2 \quad (20)$$

and $n \in \{250, 500, 1000\}$. For the local estimation of the $\lambda(x_i)$'s, we use the Epanechnikov kernel.

For each n , the criterion $CV(h)$ from (12) was minimized to estimate an optimal h . Bandwidth estimates of 24, 14, and 12 were obtained for the sample sizes 250, 500, and 1000, respectively. Figure 5 shows the plots of the estimates obtained. We then use these estimates of the bandwidths in the IGLE algorithm.

Plots of the generated data set (of size 1000) and the posteriors versus the predictors for the 2-component mixture of linear regressions model, the ME model, and the nonparametric model using the IGLE algorithm can be found in Figure 6. The IGLE algorithm provides a better estimate to the true curve than the ME EM algorithm when assuming the mixing proportions are a function of the predictors. The ME EM algorithm does not provide the flexibility to pick up the curvature in this simulation, and in fact, the ME solution looks very similar to the global λ solution.

In Table 1, we report the mean squared error (MSE) results when simulating

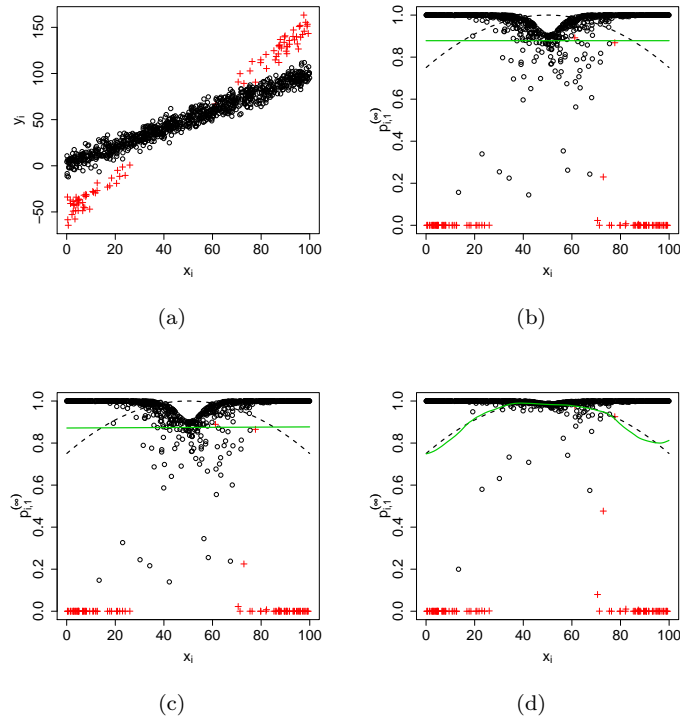


Figure 6: (a) 1000 generated values according to the model in (19). Plots of the posterior membership probabilities versus the predictors from (b) the mixture of linear regressions EM algorithm, (c) ME EM algorithm, and (d) IGLE algorithm. The black dashed curves correspond to the functional form used for the simulation and the green curves correspond to the mixing proportion estimates provided by the various methods.

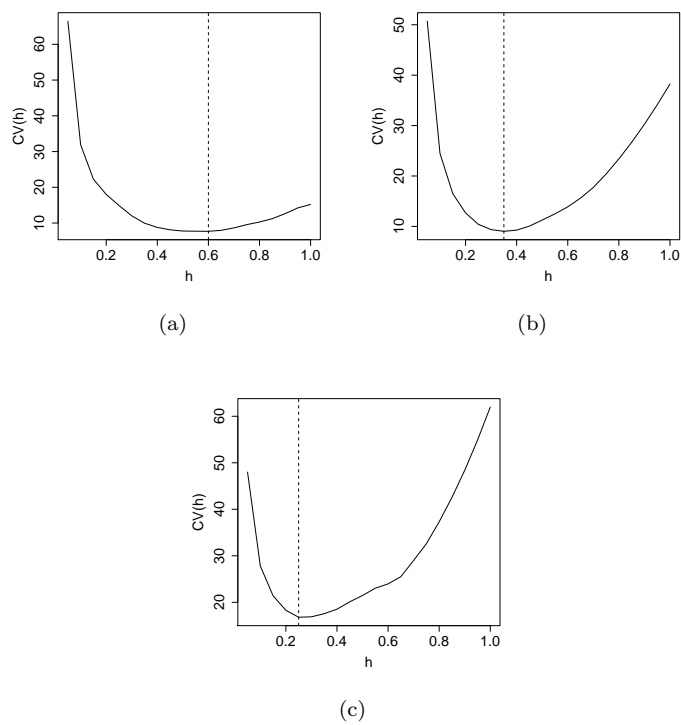


Figure 7: Plots of the cross-validation criterion $CV(h)$ versus the bandwidth (h) for sample sizes (a) 250, (b) 500, and (c) 1000 and $\lambda(x_i)$ as given in (23).

	250	500	1000
EM	0.0635	0.0627	0.0623
ME EM	0.0018	0.0009	0.0004
IGLE	0.0023	0.0017	0.0011

Table 2: Average mean squared errors for 1000 data sets for each $n \in \{250, 500, 1000\}$ using model (22).

1000 data sets for each n . The MSE is defined as

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (\widehat{\lambda}(x_i) - \lambda(x_i))^2, \quad (21)$$

where $\lambda(x_i)$ is the true value of the mixing proportion at x_i and $\widehat{\lambda}(x_i)$ is the corresponding estimate.^[10] For this simulation, the regular mixture of linear regressions EM and ME EM algorithm perform similarly while the IGLE algorithm performs much better according to the MSE.

Next we provide a simulation where the true $\lambda(x_i)$ come from an ME model. We simulate the realizations x_1, \dots, x_n from the random variable $X_i \sim Unif(49, 51)$ and the realizations y_1, \dots, y_n from the random variable

$$Y_i \sim \begin{cases} N(1000 - 10x_i, 4), & \text{with probability } \lambda(x_i); \\ N(225 + 5x_i, 5), & \text{with probability } 1 - \lambda(x_i), \end{cases} \quad (22)$$

where

$$\lambda(x_i) = \frac{\exp\{100 - 2x_i\}}{1 + \exp\{100 - 2x_i\}} \quad (23)$$

and $n \in \{250, 500, 1000\}$. For the local estimation of these $\lambda(x_i)$'s, we again use the Epanechnikov kernel. Finally, when minimizing the criterion $CV(h)$ from (12), we obtain bandwidth estimates of 0.60, 0.35, and 0.25 for the sample sizes 250, 500, and 1000, respectively. Figure 7 shows plots of the estimates obtained.

Plots of the generated data set (of size 1000) and the posteriors versus the predictors for the 2-component mixture of linear regressions model, the ME model, and the nonparametric model using the IGLE algorithm can be found in

¹⁰In the regular mixture of linear regressions EM, $\widehat{\lambda}(x_i)$ is simply the global estimate $\hat{\lambda}$.

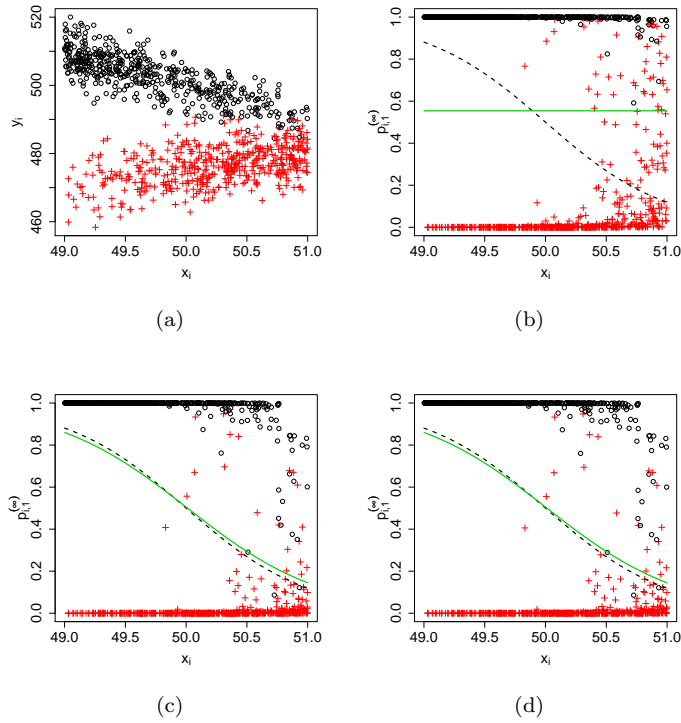


Figure 8: (a) 1000 generated values according to the model in (22). Plots of the posterior membership probabilities versus the predictors from (b) the mixture of linear regressions EM algorithm, (c) ME EM algorithm, and (d) IGLE algorithm. The black dashed curves correspond to the functional form used for the simulation and the green curves correspond to the mixing proportion estimates provided by the various methods.

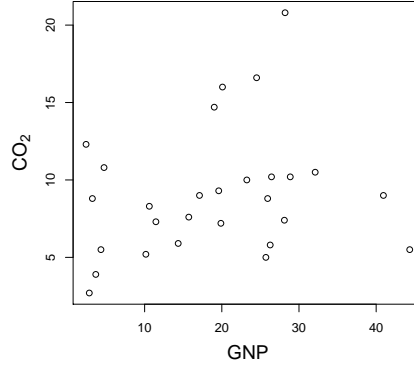


Figure 9: Plot of the CO₂ data set.

Figure 8. The ME EM algorithm and IGLE algorithm provide similar plots and detect the trend. According to the MSE values of Table 2, the ME EM model does provide the best fit. However, the IGLE algorithm performs similarly according to the MSE, thus validating the ME model.

Next, we apply all three algorithms to some real data sets and compare the various mixing proportion estimates.

4.2 CO₂ Data

The plot in Figure 9 shows a dataset consisting of the gross national product (GNP) per capita in 1996 and the estimated carbon dioxide (CO₂) emissions per capita in 1996 for a group of 28 nations. As pointed out in Hurn, Justel, and Robert (2003), these data do appear to be spread out; “however, there do seem to be several groups for which a linear model would be a reasonable approximation.” They further point out that identification of such groups could clarify the potential development paths of lower GNP countries.

A thorough study of these data from the likelihood and Bayesian perspectives can be found in Hurn *et al.* (2003), Pena *et al.* (2003), and Young (2007). These

Parameter	EM	ME EM	IGLE
β_1	8.679	8.872	8.819
σ_1	-0.023	-0.030	-0.028
β_2	2.049	2.0201	2.027
σ_2	1.415	1.493	1.476
$\ell_o(\psi)$	0.677	0.673	0.674
	0.809	0.821	0.817
	-66.940	-66.297	-66.138

Table 3: The point estimates for the CO₂ data using the regular mixture of linear regressions EM, the ME EM, and the IGLE algorithm.

works address issues such as label switching, standard error estimation, and determining the number of components. Furthermore, they all find a mixture of regressions with $m = 2$ components appropriate for this data. In other words, the model of interest to fit is

$$y_i \sim \begin{cases} \mathbf{x}_i^T \boldsymbol{\beta}_1 + \epsilon_{i,1}, & \text{with probability } \lambda; \\ \mathbf{x}_i^T \boldsymbol{\beta}_2 + \epsilon_{i,2}, & \text{with probability } 1-\lambda, \end{cases} \quad (24)$$

where the $\epsilon_{i,j} \sim N(0, \sigma_j^2)$ are independent, $i = 1, \dots, 28$ and $j = 1, 2$.

The three algorithms discussed in this paper are applied to the CO₂ data set. The model we wish to fit is the same as (24), but with λ replaced by $\lambda(x_i)$. For the IGLE algorithm, we applied the triweight kernel to the data with a bandwidth of $h = 14$. The choice of this kernel and bandwidth provided a relatively smooth curve for the estimated $\lambda(x_i)$ as well as an increase in the observed log likelihood over the other methods. The point estimates of the regression coefficients and the standard deviations as well as the observed log likelihoods can be found in Table 3. All three algorithms provide similar point estimates of these quantities. Furthermore, Figure 10 shows that both the IGLE and ME EM results suggest that countries with larger GNP are less likely to follow the steeper GNP-CO curve, a conclusion that is impossible using a standard mixture of regressions.

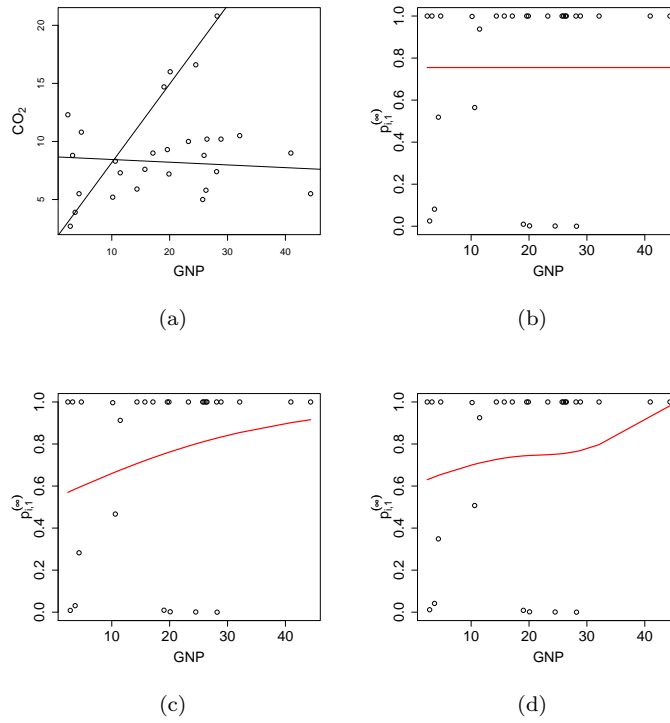
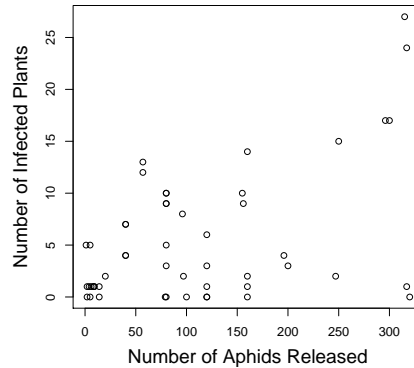


Figure 10: (a) The CO₂ data with regression lines estimated using the EM algorithm and plots of the posterior membership probabilities versus the predictors from (b) the mixture of linear regressions EM algorithm, (c) ME EM algorithm, and (d) IGLE algorithm. The red curves correspond to the mixing proportion estimates provided by the various methods.



Parameter	EM	ME EM	IGLE
β_1	0.859	0.864	0.833
σ_1	1.125	1.124	1.109
β_2	3.475	3.555	3.499
σ_2	0.055	0.055	0.055
$\ell_o(\psi)$	3.115	3.110	3.120
	-132.065	-132.017	-129.597

Table 4: The point estimates for the aphids data using the regular mixture of linear regressions EM, the ME EM, and the IGLE algorithm.

The model we wish to fit is

$$y_i \sim \begin{cases} \mathbf{x}_i^T \boldsymbol{\beta}_1 + \epsilon_{i,1}, & \text{with probability } \lambda; \\ \mathbf{x}_i^T \boldsymbol{\beta}_2 + \epsilon_{i,2}, & \text{with probability } 1-\lambda, \end{cases} \quad (25)$$

where the $\epsilon_{i,j} \sim N(0, \sigma_j^2)$ are independent, $i = 1, \dots, 51$ and $j = 1, 2$. We analyze these data using the same procedures as for the CO₂ data set. The model we wish to fit is the same as (25), but with λ replaced by $\lambda(x_i)$. For the IGLE algorithm, we again applied the triweight kernel to the data along with a bandwidth of $h = 100$. We have selected a wide bandwidth due to the relatively small sample size (as with the CO₂ data set). This produces a fairly smooth curve for the estimated $\lambda(x_i)$ as well as an increase in the observed log likelihood over the other methods. The point estimates of the regression coefficients and the standard deviations as well as the observed log likelihoods can be found in Table 4. Again, all three algorithms provide similar point estimates.

Figure 12 depicts plots of the aphids data set and the posteriors versus the predictors for the 2-component mixture of linear regressions model, the ME model, and the nonparametric model using the IGLE algorithm. The IGLE algorithm and the ME EM appear similar, but the IGLE fit has more curvature than the ME fit, indicating a cyclical trend in which the maximum value for $\lambda(x_i)$ appears to occur at $x_i = 200$. This may be of interest to the researchers because this gives a sense of the number of aphids corresponding to the greatest

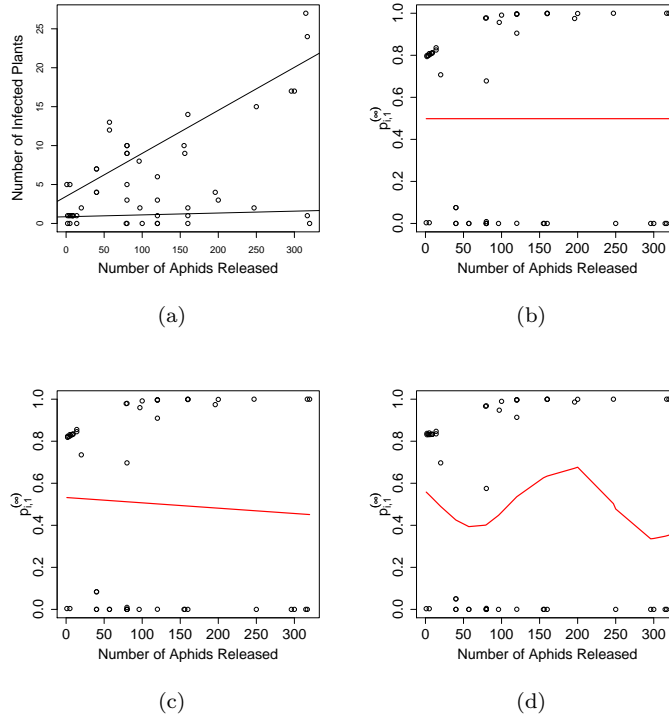


Figure 12: (a) The aphids data with regression lines estimated using the EM algorithm and plots of the posterior membership probabilities versus the predictors from (b) the mixture of linear regressions EM algorithm, (c) ME EM algorithm, and (d) IGLE algorithm. The red curves correspond to the mixing proportion estimates provided by the various methods.

probability of belonging to the “first” component. This is the component where a small number of plants are infected for all numbers of aphids released, which Turner (2000) suggests involves aphids past their maiden phase. Thus, our method gives a more nuanced view than a standard mixture of regressions.

5 Discussion

This article presents a method for allowing mixing proportions in a mixture-of-regressions model to vary as a function of the predictors. The analyses that

result are more detailed than those of a standard mixture of regressions model, as we have argued throughout. Our novel nonparametric approach provides more flexibility than the parametric mixture of experts (ME) approach, and this flexibility can either provide different-looking results than the ME method or serve to validate the ME results.

Some challenges faced by our method are the same as those faced by other local smoothing methods, such as kernel selection and bandwidth selection to provide a seemingly smooth form for $\lambda_j(\mathbf{x}_i)$. In addition to dealing with these challenges, possible future work includes developing a formal test for the efficacy of this model. Such a test would state

$$\begin{aligned} H_0 & : \lambda_j(\mathbf{x}_i) \equiv \lambda_j \quad \forall i \\ H_1 & : \lambda_j(\mathbf{x}_i) \text{ is nonconstant.} \end{aligned} \tag{26}$$

This test could be applied to other mixture of regression models beyond mixtures of linear regressions, such as mixtures of generalized linear models or mixtures of survival regressions.

A An EM Algorithm for mixtures of linear regressions

Suppose we have n independent univariate observations, y_1, \dots, y_n , each with a corresponding vector of predictors, $\mathbf{x}_1, \dots, \mathbf{x}_n$, where $\mathbf{x}_i = (x_{i,1}, \dots, x_{i,p})^T$ for $i = 1, \dots, n$. We often set $x_{i,1} = 1$ to allow for an intercept term. Let $\mathbf{y} = (y_1, \dots, y_n)^T$ and \mathbf{X} be the $n \times p$ matrix consisting of the predictor vectors. If the constant λ_j denotes the probability that an observation belongs to class j , the marginal density of y_i is $f(\mathbf{y}_i; \mathbf{x}_i, \boldsymbol{\psi})$, as given in Equation (2). Maximum likelihood estimation for the parameters in this model may be accomplished using an Expectation-Maximization (EM) algorithm (Dempster, Laird, and Rubin (1977)).

Let us denote the parameter vector by $\boldsymbol{\psi} = (\boldsymbol{\theta}_1^T, \dots, \boldsymbol{\theta}_m^T, \lambda_1, \dots, \lambda_{m-1})^T$, where $\boldsymbol{\theta}_j = (\boldsymbol{\beta}_j^T, \sigma_j^2)^T$. The log likelihood function is

$$\ell_o(\boldsymbol{\psi}) = \sum_{i=1}^n \log \sum_{j=1}^m \lambda_j (2\pi\sigma_j^2)^{-1/2} \exp\left\{-\frac{1}{2\sigma_j^2} \left(y_i - \mathbf{x}_i^T \boldsymbol{\beta}_j\right)^2\right\}. \quad (27)$$

We augment the observed data $(y_i, \mathbf{x}_i^T)^T$ by the indicator variables

$$Z_{i,j} = \mathbf{I}\{\text{observation } i \text{ belongs to component } j\},$$

$1 \leq j \leq m$. The log likelihood based on these “complete” data is then

$$\ell_c(\boldsymbol{\psi}) = \sum_{i=1}^n \sum_{j=1}^m Z_{i,j} \log \left[\lambda_j (2\pi\sigma_j^2)^{-1/2} \exp\left\{-\frac{1}{2\sigma_j^2} \left(y_i - \mathbf{x}_i^T \boldsymbol{\beta}_j\right)^2\right\} \right]. \quad (28)$$

The E-Step says for iteration t , $t = 0, 1, \dots$, compute the expected complete data log likelihood

$$\mathcal{Q}(\boldsymbol{\psi}; \boldsymbol{\psi}^{(t)}) = \sum_{i=1}^n \sum_{j=1}^m p_{i,j}^{(t)} \log \left[\lambda_j (2\pi\sigma_j^2)^{-1/2} \exp\left\{-\frac{1}{2\sigma_j^2} \left(y_i - \mathbf{x}_i^T \boldsymbol{\beta}_j\right)^2\right\} \right], \quad (29)$$

where

$$p_{i,j}^{(t)} = \frac{\lambda_j^{(t)} (2\pi\sigma_j^{2(t)})^{-1/2} \exp\left\{-\frac{1}{2\sigma_j^{2(t)}} \left(y_i - \mathbf{x}_i^T \boldsymbol{\beta}_j^{(t)}\right)^2\right\}}{\sum_{l=1}^m \lambda_l^{(t)} (2\pi\sigma_l^{2(t)})^{-1/2} \exp\left\{-\frac{1}{2\sigma_l^{2(t)}} \left(y_i - \mathbf{x}_i^T \boldsymbol{\beta}_l^{(t)}\right)^2\right\}} \quad (30)$$

$$= \left[1 + \sum_{l \neq j} \frac{\lambda_l^{(t)} \sigma_j^{(t)}}{\lambda_j^{(t)} \sigma_l^{(t)}} \exp\left\{\frac{1}{2} \left[\frac{1}{\sigma_j^{2(t)}} \left(y_i - \mathbf{x}_i^T \boldsymbol{\beta}_j^{(t)}\right)^2 - \frac{1}{\sigma_l^{2(t)}} \left(y_i - \mathbf{x}_i^T \boldsymbol{\beta}_l^{(t)}\right)^2 \right] \right\} \right]^{-1}. \quad (31)$$

The $p_{i,j}^{(t)}$ values may be called the *posterior component membership probabilities*. Note that expression (31) gives a stable formula for numerical computations, avoiding the indeterminate form 0/0 that can occur in expression (30) when using computer arithmetic.

The M-Step maximizes $\mathcal{Q}(\boldsymbol{\psi}; \boldsymbol{\psi}^{(t)})$ with respect to $\boldsymbol{\psi}$, yielding the update $\boldsymbol{\psi}^{(t+1)}$. Letting $\mathbf{W}_j^{(t)} = \text{diag}(Z_{1,j}^{(t)}, \dots, Z_{n,j}^{(t)})$, $\boldsymbol{\psi}^{(t+1)}$ consists of

$$\lambda_j^{(t+1)} = \frac{1}{n} \sum_{i=1}^n Z_{i,j}^{(t)}, \quad (32)$$

$$\boldsymbol{\beta}_j^{(t+1)} = (\mathbf{X}^T \mathbf{W}_j^{(t)} \mathbf{X})^{-1} \mathbf{X}^T \mathbf{W}_j^{(t)} \mathbf{y}, \quad (33)$$

and

$$\sigma_j^{2(t+1)} = \frac{\left\| \mathbf{W}_j^{1/2(t)} (\mathbf{y} - \mathbf{X}^T \boldsymbol{\beta}_j^{(t+1)}) \right\|^2}{\text{tr}(\mathbf{W}_j^{(t)})}, \quad (34)$$

where $\|\mathbf{v}\|^2 = \mathbf{v}^T \mathbf{v}$ and $\text{tr}(\mathbf{A})$ means the trace of the matrix \mathbf{A} . Iterate equations (32) through (34) until the stopping criterion of $\ell_o(\boldsymbol{\psi}^{(t+1)}) - \ell_o(\boldsymbol{\psi}^{(t)}) < \epsilon$ is attained for small $\epsilon > 0$.

B An EM Algorithm for HME models

First, define the indicator random variable

$$Z_{i,j,l} = \mathbb{I}\{\text{observation } i \text{ belongs to node } (j,l)\}. \quad (35)$$

Then, the E-Step for iteration t , $t = 0, 1, \dots$, computes the expected complete data log likelihood

$$\begin{aligned} \mathcal{Q}(\boldsymbol{\psi}; \boldsymbol{\psi}^{(t)}) &= \sum_{i=1}^n \sum_{j=1}^{m_1} \sum_{l=1}^{m_2} p_{i,j,l}^{(t)} \log \left[\lambda_j(\mathbf{x}_i, \boldsymbol{\tau}) \lambda_{j,l}(\mathbf{x}_i, \boldsymbol{\omega}_j) (2\pi\sigma_{j,l}^2)^{-1/2} \right. \\ &\quad \left. \times \exp \left\{ -\frac{1}{2\sigma_{j,l}^2} \left(y_i - \mathbf{x}_i^T \boldsymbol{\beta}_{j,l} \right)^2 \right\} \right], \quad (36) \end{aligned}$$

where

$$\begin{aligned} p_{i,j,l}^{(t)} &= \frac{\frac{\lambda_j(\mathbf{x}_i, \boldsymbol{\tau}^{(t)}) \lambda_{j,l}(\mathbf{x}_i, \boldsymbol{\omega}_j^{(t)})}{\sqrt{2\pi\sigma_{j,l}^2}} \exp \left\{ -\frac{1}{2\sigma_{j,l}^2} \left(y_i - \mathbf{x}_i^T \boldsymbol{\beta}_{j,l}^{(t)} \right)^2 \right\}}{\sum_{j^*=1}^{m_1} \sum_{l^*=1}^{m_2} \frac{\lambda_{j^*}(\mathbf{x}_i, \boldsymbol{\tau}^{(t)}) \lambda_{j^*,l^*}(\mathbf{x}_i, \boldsymbol{\omega}_{j^*}^{(t)})}{\sqrt{2\pi\sigma_{j^*,l^*}^2}} \exp \left\{ -\frac{1}{2\sigma_{j^*,l^*}^2} \left(y_i - \mathbf{x}_i^T \boldsymbol{\beta}_{j^*,l^*}^{(t)} \right)^2 \right\}} \\ &= \left[1 + \sum_{(j^*,l^*) \neq (j,l)} \frac{\lambda_{j^*}(\mathbf{x}_i, \boldsymbol{\tau}^{(t)}) \lambda_{j^*,l^*}(\mathbf{x}_i, \boldsymbol{\omega}_{j^*}^{(t)})}{\lambda_j(\mathbf{x}_i, \boldsymbol{\tau}^{(t)}) \lambda_{j,l}(\mathbf{x}_i, \boldsymbol{\omega}_j^{(t)})} \frac{\sigma_{j,l}^{(t)}}{\sigma_{j^*,l^*}^{(t)}} \right. \\ &\quad \left. \times \exp \left\{ \frac{1}{2} \left[\frac{1}{\sigma_{j,l}^2} \left(y_i - \mathbf{x}_i^T \boldsymbol{\beta}_{j,l}^{(t)} \right)^2 - \frac{1}{\sigma_{j^*,l^*}^2} \left(y_i - \mathbf{x}_i^T \boldsymbol{\beta}_{j^*,l^*}^{(t)} \right)^2 \right] \right\} \right]^{-1}. \end{aligned}$$

The M-Step maximizes $Q(\boldsymbol{\psi}; \boldsymbol{\psi}^{(t)})$ with respect to $\boldsymbol{\psi}$, yielding the update $\boldsymbol{\psi}^{(t+1)}$. Letting $\mathbf{W}_{j,l}^{(t)} = \text{diag}(p_{1,j,l}^{(t)}, \dots, p_{n,j,l}^{(t)})$, $\boldsymbol{\psi}^{(t+1)}$ includes

$$\boldsymbol{\beta}_{j,l}^{(t+1)} = (\mathbf{X}_n^T \mathbf{W}_{j,l}^{(t)} \mathbf{X}_n)^{-1} \mathbf{X}_n^T \mathbf{W}_{j,l}^{(t)} \mathbf{y} \quad (37)$$

and

$$\sigma_{j,l}^{2(t+1)} = \frac{\left\| \mathbf{W}_{j,l}^{1/2(t)} (\mathbf{y} - \mathbf{X}_n^T \boldsymbol{\beta}_{j,l}^{(t+1)}) \right\|^2}{\text{tr}(\mathbf{W}_{j,l}^{(t)})}. \quad (38)$$

The updates $\boldsymbol{\tau}_j^{(t+1)}$ and $\boldsymbol{\omega}_{j,l}^{(t+1)}$ are found by solving

$$\frac{\partial b_1(\boldsymbol{\tau}_j)}{\partial \boldsymbol{\tau}_j} = \mathbf{0}, \quad (39)$$

where

$$b_1(\boldsymbol{\tau}_j) = \sum_{i=1}^n \sum_{j=1}^{m_1} \sum_{l=1}^{m_2} p_{i,j,l}^{(t)} \log \lambda_j(\mathbf{x}_i, \boldsymbol{\tau}),$$

and

$$\frac{\partial b_2(\boldsymbol{\omega}_{j,l})}{\partial \boldsymbol{\omega}_{j,l}} = \mathbf{0}, \quad (40)$$

where

$$b_2(\boldsymbol{\omega}_{j,l}) = \sum_{i=1}^n \sum_{j=1}^{m_1} \sum_{l=1}^{m_2} p_{i,j,l}^{(t)} \log \lambda_{j,l}(\mathbf{x}_i, \boldsymbol{\omega}_j).$$

Both (39) and (40) can be solved by an iteratively reweighted least squares (IRLS) routine (McCullagh and Nelder (1989)).

In order to work out the IRLS routine for (39), we let $\mathbf{W}_{j,\cdot}^{(t)} = \sum_{l=1}^{m_1} \mathbf{W}_{j,l}^{(t)}$, \mathbf{Q}_1 be an $n \times n$ diagonal matrix with i^{th} diagonal entry equal to $\lambda_j(\mathbf{x}_i, \boldsymbol{\tau}^{(t,s)})(1 - \lambda_j(\mathbf{x}_i, \boldsymbol{\tau}^{(t,s)}))$, and $\mathbf{p}_{1,j} = (\lambda_j(\mathbf{x}_1, \boldsymbol{\tau}^{(t,s)}), \dots, \lambda_j(\mathbf{x}_n, \boldsymbol{\tau}^{(t,s)}))^T$. Then, the Newton-Raphson update is

$$\begin{aligned} \boldsymbol{\tau}_j^{(t,s+1)} &= \boldsymbol{\tau}_j^{(t,s)} - \left(\frac{\partial^2 b_1(\boldsymbol{\tau}_j)}{\partial \boldsymbol{\tau}_j \partial \boldsymbol{\tau}_j^T} \Big|_{\boldsymbol{\tau}_j^{(t,s)}} \right)^{-1} \frac{\partial b_1(\boldsymbol{\tau}_j)}{\partial \boldsymbol{\tau}_j} \Big|_{\boldsymbol{\tau}_j^{(t,s)}} \\ &= \boldsymbol{\tau}_j^{(t,s)} + (\mathbf{X}_n^T \mathbf{W}_{j,\cdot}^{(t)} \mathbf{Q}_1 \mathbf{X}_n)^{-1} \mathbf{X}_n^T \mathbf{W}_{j,\cdot}^{(t)} \mathbf{p}_{1,j} \\ &= (\mathbf{X}_n^T \mathbf{W}_{j,\cdot}^{(t)} \mathbf{Q}_1 \mathbf{X}_n)^{-1} \mathbf{X}_n^T \mathbf{W}_{j,\cdot}^{(t)} \mathbf{Q}_1 (\mathbf{X}_n \boldsymbol{\tau}_j^{(t,s)} + \mathbf{Q}_1^{-1} \mathbf{p}_{1,j}) \\ &= (\mathbf{X}_n^T \mathbf{W}_j^* \mathbf{X}_n)^{-1} \mathbf{X}_n^T \mathbf{W}_j^* \mathbf{v}_{1,j}, \end{aligned} \quad (41)$$

where (41) is a WLS estimate. We run these equations for $s = 1, 2, \dots$ until a convergence criterion is satisfied and let $\boldsymbol{\tau}_j^{(t+1)}$ equal $\boldsymbol{\tau}_j^{(t,s+1)}$.

The IRLS routine for (40) is similar. Let \mathbf{Q}_2 be an $n \times n$ diagonal matrix with i^{th} diagonal entry equal to $\lambda_{j,l}(\mathbf{x}_i, \boldsymbol{\omega}_j^{(t,s)})(1 - \lambda_{j,l}(\mathbf{x}_i, \boldsymbol{\omega}_j^{(t,s)}))$ and $\mathbf{p}_{2,j} = (\lambda_{j,l}(\mathbf{x}_1, \boldsymbol{\omega}_j^{(t,s)}), \dots, \lambda_{j,l}(\mathbf{x}_n, \boldsymbol{\omega}_j^{(t,s)}))^{\text{T}}$. As in (41), the Newton-Raphson update is

$$\begin{aligned} \boldsymbol{\omega}_{j,l}^{(t,s+1)} &= (\mathbf{X}_n^{\text{T}} \mathbf{W}_{j,l}^{(t)} \mathbf{Q}_2 \mathbf{X}_n)^{-1} \mathbf{X}_n^{\text{T}} \mathbf{W}_{j,l}^{(t)} \mathbf{Q}_2 (\mathbf{X}_n \boldsymbol{\omega}_{j,l}^{(t,s)} + \mathbf{Q}_2^{-1} \mathbf{p}_{2,j}) \\ &= (\mathbf{X}_n^{\text{T}} \mathbf{W}_j^{\dagger} \mathbf{X}_n)^{-1} \mathbf{X}_n^{\text{T}} \mathbf{W}_j^{\dagger} \mathbf{v}_{2,j}. \end{aligned} \quad (42)$$

Run these equations for $s = 1, 2, \dots$ until a convergence criterion is satisfied and let $\boldsymbol{\omega}_{j,l}^{(t+1)}$ equal $\boldsymbol{\omega}_{j,l}^{(t,s+1)}$.

Finally, iterate the entire algorithm (which includes the IRLS loops) until the stopping criterion $\ell_o(\boldsymbol{\psi}^{(t+1)}) - \ell_o(\boldsymbol{\psi}^{(t)}) < \epsilon$ is attained for $\epsilon > 0$. Details for an EM algorithm corresponding to the model with a general number of hierarchies (as well as convergence results) can be found in Jordan and Xu (1995).

References

- Cohen E (1980). *Inharmonic Tone Perception*. Ph.D. thesis, Stanford University. Unpublished.
- Dempster AP, Laird NM, Rubin DB (1977). “Maximum Likelihood from Incomplete Data via the EM Algorithm.” *Journal of the Royal Statistical Society, Series B*, **39**(1), 1–38.
- Hastie T, Tibshirani R, Friedman JH (2001). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, New York.
- Huang M (2006). *Nonparametric Mixture of Regressions Model*. Master’s thesis, The Pennsylvania State University. Unpublished.

- Hurn M, Justel A, Robert CP (2003). “Estimating Mixtures of Regressions.” *Journal of Computational and Graphical Statistics*, **12**(1), 55–79.
- Jacobs RA, Jordan MI, Nowlan SJ, Hinton GE (1991). “Adaptive Mixtures of Local Experts.” *Neural Computation*, **3**(1), 79–87.
- Jacobs RA, Peng F, Tanner MA (1997). “A Bayesian Approach to Model Selection in Hierarchical Mixtures-of-Experts Architectures.” *Neural Networks*, **10**(2), 231–241.
- Jordan MI, Jacobs RA (1992). “Hierarchies of Adaptive Experts.” In J Moody, S Hanson, R Lippmann (eds.), “Nueral Information Processing Systems 4,” pp. 985–993. San Mateo, CA: Morgan Kaufmann.
- Jordan MI, Jacobs RA (1994). “Hierarchical Mixtures of Experts and the EM Algorithm.” *Neural Computations*, **6**, 181–214.
- Jordan MI, Xu L (1995). “Convergence Results for the EM Approach to Mixtures of Experts Architectures.” *Neural Networks*, **8**(9), 1409–1431.
- Justel A, Pena D (1996). “Gibbs Sampling Will Fail in Outlier Problems with Strong Masking.” *Journal of Computational and Graphical Statistics*, **5**(2), 176–189.
- McCullagh P, Nelder JA (1989). *Generalized Linear Models*. Chapman & Hall/CRC, Florida, 2nd edition.
- Nadaraya EA (1964). “On Estimating Regression.” *Theory of Probability and Its Applications*, **9**(1), 141–142.
- Nagin DS (1999). “Analyzing Developmental Trajectories: A Semiparametric Group-Based Approach.” *Psychological Methods*, **4**(2), 139–157.

- Pena D, Rodriguez J, Tiao GC (2003). “Identifying Mixtures of Regression Equations by the SAR Procedure.” In JM Bernardo, MJ Bayarri, JO Berger, AP Dawid, D Heckerman, AFM Smith, M West (eds.), “Bayesian Statistics 7,” pp. 327–348. Oxford: Clarendon Press.
- Quandt RE (1972). “A New Approach to Estimating Switching Regressions.” *Journal of the American Statistical Association*, **67**(338), 306–310.
- Quandt RE, Ramsey JB (1978). “Estimating Mixtures of Normal Distributions and Switching Regressions.” *Journal of the American Statistical Association*, **73**(364), 730–738.
- R Development Core Team (2008). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL <http://www.R-project.org>.
- Turner TR (2000). “Estimating the Propagation Rate of a Viral Infection of Potato Plants via Mixtures of Regressions.” *Applied Statistics*, **49**(3), 371–384.
- Turner TR (2004). *The mixreg Package: Functions to Fit Mixtures of Regressions*. R Package Version 0.0.1.
- Viele K, Tong B (2002). “Modeling with Mixtures of Linear Regressions.” *Statistics and Computing*, **12**(4), 315–330.
- Wand MP, Jones MC (1995). *Kernel Smoothing*. Chapman & Hall/CRC, Florida.
- Watson GS (1964). “Smooth Regression Analysis.” *Sankhyā - The Indian Journal of Statistics*, **26**(4), 359–372.

- Yau KKW, Lee AH, Ng ASK (2003). “Finite Mixture Regression Model with Random Effects: Application to Neonatal Hospital Length of Stay.” *Computational Statistics and Data Analysis*, **41**(3/4), 359–366.
- Young DS (2007). *A Study of Mixtures of Regressions*. Ph.D. thesis, The Pennsylvania State University. Unpublished.
- Young DS, Benaglia T, Chauveau D, Hunter DR, Elmore RT, Xuan F, Hettmansperger TP, Thomas H (2008). *The mixtools Package: Tools for Mixture Models*. R Package Version 0.3.0.