Grouped mixture of regressions

Haidar Almohri, PhD Candidate
Ratna Babu Chinnam, Ph.D., Professor
Arash Ali Amini, Ph.D, Professor*
Department of Industrial and Systems Engineering
Wayne State University, Detroit, MI 48201, U.S.A.
{*Haidar.Almohri, Ratna.Chinnam*}*@wayne.edu*

November 28, 2017

Abstract

The text of your abstract. 200 or fewer words.

Keywords: Finite mixture models, Finite mixture of linear regressions, Mixture models with must-link constraint, Expectation Maximization

*The authors gratefully acknowledge the support of *Urban Science* for sponsoring this research.

1 Introduction

One of the challenges in modeling certain populations is that the observations might be drawn from different distributions/processes underlying the overall population. In such cases, a "single" model may fail to efficiently represent the sample data and therefore the accuracy and reliability of the model might suffer. This problem has been identified more than hundred years ago (Newcomb, 1886; Pearson, 1894) and "mixture" models were introduced in order to better account for the unobserved heterogeneity in the population. Since those early days, a lot of effort has gone into developing new methodologies and to further improve the modeling. In recent years, due to increasing availability and diversity of data, the topic has experienced an increasing attention by researchers. Mixture models have been successfully employed in a variety of diverse applications such as speech recognition (Reynolds et al., 1995), image retrieval (Permuter et al., 2003), term structure modeling (Lemke, 2006), biometric verification (Stylianou et al., 2005), and market segmentation (Tuma and Decker, 2013).

Among the family of mixture models, the finite mixture of linear regression (FMR) models have been particularly popular in various fields and applications (Bierbrauer et al., 2004; Andrews and Currim, 2003; Bar-Shalom, 1978), mainly because of the advantages of linear models such as simplicity, interpretability, and scientific acceptance. In FMR, it is assumed that the distribution of the data can be represented using a convex combination of a finite (K) number of linear regression models. Equivalently, each observation belongs to one the K classes, and given the class membership, it follows the regression model associated with that class. The difficulty is that the class memberships are not known in advance.

Assuming that the dataset consists of n observations $(y_i, x_i), i = 1, \dots n$, let y_i denote the value of response variable for the i^{th} observation, and x_i the corresponding $p \times 1$ vector of independent variables (for brevity, we exclude the intercept from the notation). Let $y = (y_1, y_2, \dots, y_n) \in \mathbb{R}^n$ be the response vector, and $X = (x_1, x_2, \dots, x_n) \in \mathbb{R}^{n \times p}$ the design matrix. The whole population in this case can be represented as: AA: this is not quite correct,... but not sure if it is necessary to give the details here. We will use different notation for the grouped data later on. HA: Do you suggest to remove this?

$$Y = \sum_{k=1}^{K} \alpha_k (X^T \beta_k) + \varepsilon_k \qquad k = 1, ..., K \tag{1}$$

where Y is the $n * 1$ vector of response variables, X is the covariate matrix, β_k is the regression coefficient of the k^{th} model, K is the number of linear regression models (a.k.a. components), ϵ_k is the standard error of the k^{th} regression, and α_k is the mixture probability (the proportion of k^{th} component with respect to the total population; $\sum_{k=1}^{K} \alpha_k = 1$). We assume that K is known and $\epsilon_k \sim \mathcal{N}(0, \sigma_k^2)$. The ultimate objective is to estimate the parameters of the mixture model. In the case of FMR, the parameters to be estimated are: $\Theta = (\pi_1, \ldots, \pi_K, \beta_1, \ldots, \beta_k, \sigma_1, \ldots, \sigma_k)$.

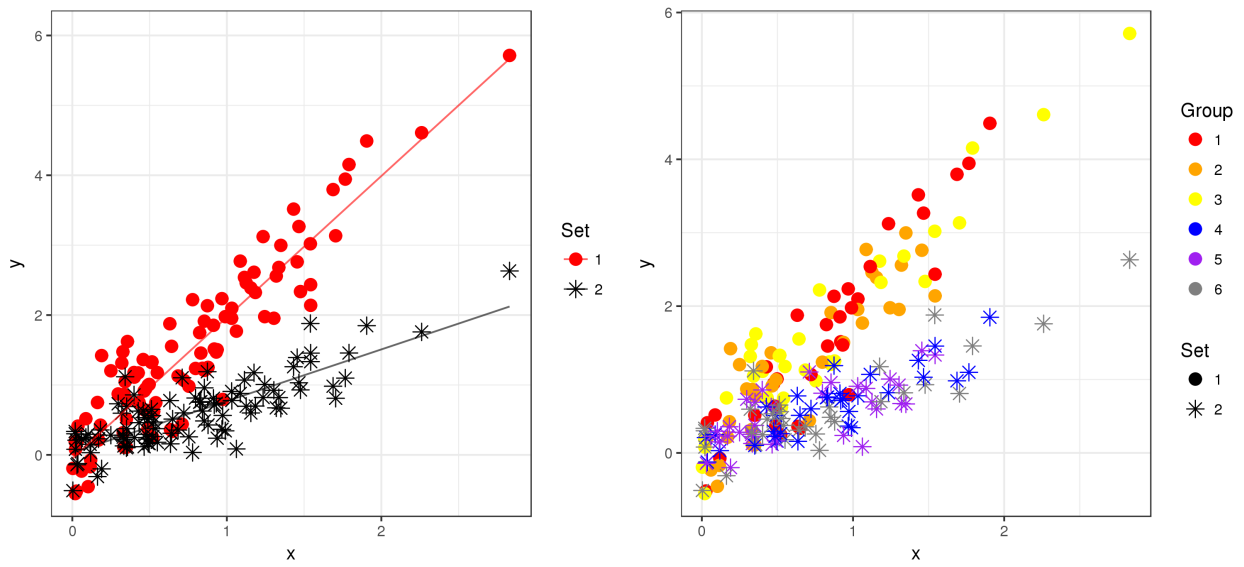
1.1 Estimating the Parameters for Mixture Models

While the parameter estimation in mixture models has been studied mainly from a likelihood point of view (De Veaux, 1989), (Quandt and Ramsey, 1978) used a moment generating function for estimating the parameters. However, maximum likelihood approach using expectation maximization (EM) (Dempster et al., 1977) remains the most widely used technique for estimating the parameters of FMR. EM approach tries to maximize the likelihood in a way that in each iteration, it is guaranteed that the value of likelihood increases. Other algorithms such as stochastic EM (Celeux and Diebolt, 1985) and classification EM (Celeux and Govaert, 1992) have been introduced as an attempt to improve the performance of the EM algorithm (see (Faria and Soromenho, 2010)). Others have employed Gibbs sampler (Diebolt and Robert, 1994)), and Bayesian approach for estimation (Hurn et al., 2003). (Chaganty and Liang, 2013) employed low-rank regression with a tensor power method as an alternative to EM algorithm for estimating the parameters.

1.2 FMR with Group Structure

Under regular FMR, the outcome is a mixture model that provides class membership for each observation of the dataset along with probability (proportion) for each component and the parameters of the model. This results in (soft) clustering the different observations into K clusters, assuming K components are employed. In some applications however, instead of individual observations, groups of observations need to be clustered or associated with the same component. For example, if FMR is being employed to model data from a retail chain, it might be necessary to associate all observations stemming from any single store to the same component. This problem is similar to what is known as "clustering with must-link constraint", which is introduced by Wagstaff and Cardie (2000) in the literature (Wagstaff et al., 2001). The main idea is to utilize experts domain

knowledge prior to clustering process in order to obtain desired properties from the clustering solution. Figure 1 illustrates the concept. The data points are synthetically generated using two components: $y_1 = \frac{1}{2}x + \epsilon_1$ and $y_2 = \frac{3}{4}x + \epsilon_2$, where $x \sim \mathcal{N}(0, 1)$, $\epsilon_1 \sim \mathcal{N}(0, 0.5)$, and $\epsilon_2 \sim \mathcal{N}(0, 0.3)$. Figure 1a shows the linear relationship between the two groups (y_1 and y_2), without any grouping (must-link) structure. In figure 1b, the data points are linked to create six (6) groups (groups 1-3 belong to y_1 and groups 4-6 belong to y_2). The data points with the same color refer to the same group. The desired outcome is that all the data points in the same group end up having the same class membership. See Basu (2009) for an extensive review of constrained clustering algorithms and applications.



(a) (b)

Figure 1: FMR with "group" constraint: (a) Synthetic, two component FMR without any constraint. (b) The same data points being divided into six groups where each group has to retain its data points.

To the best of our knowledge, all the existing algorithms have solved the problem of clustering with group structure in unsupervised/semi-supervised settings, meaning that the observations lack (or partially lack) the dependent variable. In other words, we could not find any work that addresses FMR with group structure; contribution of this manuscript.

1.3 Grouped mixture of regression

We assume that the observations belong to R *known groups*, denoted with labels $[R] := \{1, \ldots, R\}$. In each group $r \in [R]$, we observe n_r samples $(y_{ri}, x_{ri}), i = 1, \ldots, n_r$ where $y_{ri} \in \mathbb{R}$ is the response variable and $x_{ri} \in \mathbb{R}^p$ is the vector of covariates or features. We will write x_{rij} to denote the j^{th} feature in the feature vector x_{ri}. For the most part, we will treat x_{ri} as deterministic observations, i.e., we have fixed design regression models.

We assume that there are K latent (unobserved) clusters such that all the observations in group r belong to that cluster. Thus, we can assign a cluster membership variable $z_r \in \{0, 1\}^K$ to each group $r \in [R]$. We will have $z_{rk} = 1$ iff group r belongs to cluster k. With some abuse of notation, we also write $z_r = k$ in place of $z_{rk} = 1$. Given the cluster membership variable z_r, we assume that the group r observations are independent draws from a Gaussian linear regression model with parameters specified by z_r, that is,

$$p(y_{ri} \,|\, z_r = k) \overset{\text{indept}}{\sim} N(\beta_k^T x_{ri}, \sigma_r^2), \; i = 1, \ldots, n_r, \tag{2}$$

where $\beta_k \in \mathbb{R}^p$ is the coefficient vector the k^{th} regression model and σ_r^2 is the noise variance for group r. Note that we are assuming that the noise level only depends on the group and not on the underlying cluster. AA: Might need some justification. HA: How can we justify? We write $\beta = (\beta_1 \mid \cdots \mid \beta_K) \in \mathbb{R}^{p \times K}$ and $\sigma^2 = (\sigma_1^2, \ldots, \sigma_R^2) \in \mathbb{R}^R$.

As is common in mixture modeling, we assume that z_r follows a multinomial prior with parameter $\pi = (\pi_k)$, that is, $\mathbb{P}(z_r = k) = \pi_k$ for $k \in [K]$, and $z_1, \ldots, z_R$ are drawn independently. The joint distribution of y_r and z_r is then given by:

$$p_\theta(y_r, z_r) = p_\theta(z_r) \prod_{i=1}^{n_r} p_\theta(y_{ri} \,|\, z_r) = \prod_{k=1}^{K} \Big[\pi_k \prod_{i=1}^{n_r} p_\theta(y_{ri} \,|\, z_r = k) \Big]^{z_{rk}} \tag{3}$$

where we have let $\theta = (\beta, \pi, \sigma^2)$ collect all the parameters of the model. From (2), we have $p_\theta(y_{ri} \,|\, z_r = k) = \phi\big((y_{ri} - \beta_k^T x_{ri})/\sigma_r\big)$, where $\phi(\cdot)$ is the density of the standard Gaussian distri-

bution. Therefore, the so-called complete likelihood of θ given (z, y) is:

$$L(\theta \,|\, y, z) = p_\theta(y, z) = \prod_{r=1}^{R} p_\theta(y_r, z_r) = \prod_{r=1}^{R} \prod_{k=1}^{K} \Big[\underbrace{\pi_k \prod_{i=1}^{n_r} \phi\Big(\frac{y_{ri} - \beta_k^T x_{ri}}{\sigma_r}\Big)}_{=:\, \gamma_{rk}(\theta)} \Big]^{z_{rk}} \tag{4}$$

The parameter $\gamma_{rk}(\theta)$ in (4) is proportional (in k) to the posterior probability of z_r given the observation y_r, that is, $p_\theta(z_r = k \,|\, y_r) \propto_k p_\theta(y_r, z_r = k) = \gamma_{rk}(\theta)$. By normalizing $\gamma_{rk}(\theta)$ over k, we obtain the *posterior probability of cluster assignments*:

$$p_\theta(z_r = k \,|\, y_r) = \frac{\gamma_{rk}(\theta)}{\sum_{k'} \gamma_{rk'}(\theta)} =: \tau_{rk}(\theta), \tag{5}$$

for any $k \in [K]$ and $r \in [R]$. We note that the overall posterior factorizes over groups, i.e., $p_\theta(z \,|\, y) = \prod_r p_\theta(z_r \,|\, y_r)$, so it is enough to specify it for each pair z_r and y_r. Thus, $\tau_{rk}(\theta)$ is the posterior probability that group r belongs to cluster k, given all the observations y. These posterior probabilities are key estimation objectives.

An estimate $\widehat{\theta} = (\widehat{\beta}, \widehat{\phi}, \widehat{\sigma}^2)$ of θ can be obtained by maximizing (4). The classical approach to performing such optimization is by the Expectation Maximization (EM) algorithm, the details of which will be given in Section ?. Once we have the estimate $\widehat{\theta}$ of the parameters, we can calculate an estimate of the posterior probabilities as $\tau_{rk}(\widehat{\theta})$.

1.3.1 Prediction

Now assume that we have new test data point $(y_{r,\text{new}}, x_{r,\text{new}})$ in group r, for which we observe only the feature vector $x_{r,\text{new}}$ and would like to predict $y_{r,\text{new}}$. Let $(y^{\text{train}}, x^{\text{train}})$ denote all the observations used in the training. The common link between the training and test data points are the latent variables $z_1, \ldots, z_R$. In other words, since we already have a good estimate of the membership of group r based on the training data (via the posterior (5)), we can get a much better prediction of $y_{r,\text{new}}$ than what the prior model suggests. More precisely, we have the following *predictive density* for $y_{r,\text{new}}$ based on y^{train},

$$p_\theta(y_{r,\text{new}} \,|\, y^{\text{train}}) = \sum_{z_r} p_\theta(y_{r,\text{new}} \,|\, z_r)\; p_\theta(z_r \,|\, y^{\text{train}}).$$

Since, $p_\theta(z_r = k \,|\, y^{\text{train}}) = p_\theta(z_r = k \,|\, y_r^{\text{train}}) = \tau_{rk}(\theta)$, we obtain the following estimate of the predictive density

$$p_{\widehat{\theta}}(y_{r,\text{new}} \,|\, y^{\text{train}}) = \sum_{k=1}^{K} p_\theta(y_{r,\text{new}} \,|\, z_r = k)\, \tau_{rk}(\widehat{\theta}) = \sum_{k=1}^{K} \tau_{rk}(\widehat{\theta})\, \phi\Big(\frac{y_{r,\text{new}} - \widehat{\beta}_k^T x_{r,\text{new}}}{\widehat{\sigma}_r}\Big). \tag{6}$$

Note that $\widehat{\theta}$ is our estimate of the parameters based on the training data $(y^{\text{train}}, x^{\text{train}})$. In particular, the posterior mean based on (6) is $\sum_{k=1}^{K} \tau_{rk}(\widehat{\theta})\, \widehat{\beta}_k^T x_{r,\text{new}}$ which serves as the maximum a posterior (MAP) prediction for $y_{r,\text{new}}$.

1.4 Estimation

Let us now derive the EM updates for the model. Recalling (4), the complete log-likelihood of the model is

$$\ell(\theta \,|\, y, z) = \log p_\theta(y, z) = \sum_{r=1}^{R} \sum_{k=1}^{K} z_{rk} \Big[\log \pi_k + \sum_{i=1}^{n_r} \log \phi\Big(\frac{y_{ri} - \beta_k^T x_{ri}}{\sigma_r}\Big)\Big]. \tag{7}$$

Treating the class latent memberships $\{z_r\}$ as missing data, we perform the EM updates to simultaneously estimate $\{z_r\}$ and θ:

E-Step: We replace (7) with its expectation under the approximate posterior of $\{z_r\}$:

$$\begin{aligned} F(\theta; \widehat{\theta}) := E_{z \sim \tau(\widehat{\theta})}[\ell(\theta \,|\, y, z)] &= \sum_{r=1}^{R} \sum_{k=1}^{K} \tau_{rk}(\widehat{\theta}) \Big[\log \pi_k + \sum_{i=1}^{n_r} \log \phi\Big(\frac{y_{ri} - \beta_k^T x_{ri}}{\sigma_r}\Big)\Big] \\ &= \sum_{k=1}^{K} \tau_{+k}(\widehat{\theta}) \log \pi_k + \sum_{r=1}^{R} \sum_{k=1}^{K} \sum_{i=1}^{n_r} \tau_{rk}(\widehat{\theta}) \log \phi\Big(\frac{y_{ri} - \beta_k^T x_{ri}}{\sigma_r}\Big) \end{aligned} \tag{8}$$

where $\tau_{rk}(\theta)$ is the posterior given in (5), $\tau_{+k}(\theta) = \sum_r \tau_{rk}(\theta)$.

M-Step: We then maximize $F(\theta; \widehat{\theta})$ over θ, giving the update rule for the parameters $\theta = (\beta, \pi, \sigma^2)$.

To derive the update rules, we maximize $F(\theta; \widehat{\theta})$ by a sequential block coordinate ascent, in each step maximizing over one of the three sets of parameters π, β and σ^2, while fixing the others. The updates are summarized in Algorithm 1. The details can be found in Appendix A.1.

Algorithm 1 Grouped mixture of regression (GMR)

1: Compute feature covariances for each group: $\widehat{\Sigma}_r \leftarrow \frac{1}{n_r}\sum_{i=1}^{n_r} x_{ri}x_{ri}^T$

2: Compute feature-response cross-covariances: $\widehat{\rho}_r \leftarrow \frac{1}{n_r}\sum_{i=1}^{n_r} y_{ri}x_{ri}$

3: For any class posterior $\tau = (\tau_{rk})$ define the following weights:

$$\tau_{+k}(\tau) := \sum_r \tau_{rk}, \quad w_{rk}(\tau) := n_r\tau_{rk}, \quad w_{+k}(\tau) := \sum_r w_{rk}, \quad \check{w}_{rk}(\tau) := \frac{w_{rk}}{w_{+k}}.$$

and the weighted covariances: $\widetilde{\Sigma}_k(\tau) := \sum_{r=1}^R \check{w}_{rk}\widehat{\Sigma}_r$ and $\widetilde{\rho}_k(\tau) := \sum_{r=1}^R \check{w}_{rk}\widehat{\rho}_r$.

4: For any parameter $\theta = (\pi, \beta, \sigma^2)$ and class posterior $\tau = (\tau_{rk})$, define the errors:

$$E_{rk}(\beta) := \frac{1}{n_r}\sum_i^{n_r}(y_{ri} - \beta_k^T x_{ri})^2, \quad \overline{E}_k(\beta, \tau) := \sum_r \check{w}_{rk}(\tau)E_{rk}(\beta)$$

5: **while** not converged **do**

6: Update class frequencies: $\pi_k \leftarrow \tau_{+k}(\tau)/R, \quad k \in [K]$

7: Update regression coefficients: $\beta_k \leftarrow \widetilde{\Sigma}_k^{-1}(\tau)\,\widetilde{\rho}_k(\tau), \quad k \in [K]$

8: Update noise variances: $\sigma_k^2 \leftarrow \overline{E}_k(\beta, \tau), \quad k \in [K]$

9: **end while**

2 Empirical analysis

AA: Needs a little summary of what is going on in this section

AA: Must comment somewhere that we turn soft labels to hard labels by MAP rule.

2.1 Synthetic data simulations

To evaluate the effectiveness of the EM algorithm for FMR with group structure constraints, we employ Monte Carlo simulation experiments.

Experiment setup. We generate the synthetic data from the GMR model (2) with a random design where we generate the feature vectors by drawing each $x_i \sim N(0, \Sigma)$. HA: Σ is drawn from a normalized Wishart distribution. AA: Should mention how we choose Σ. I prefer drawing it from the Wishart distribution, but it might be different in your simulations. Recall that K is the number of clusters (or mixture components) and R the number of groups. We will use equal number of observations per group, that is, n_r is the same for all $r = 1, \ldots, R$. Letting $n = \sum_{r=1}^R n_r$

Table 1: Monte Carlo Simulation Parameters

K	d	G_k	n	Noise Level (σ_r)	β-distance (δ_β)
2	2	10	(100, 200, 400, 800)	(2, 4, 6, 8, 10)	(4, 7, 11)
4	(2, 4)				

be the total number of observation, we will have $n_r = n/R = R/K$ in that case. Let G_k be the number of groups in cluster k. In general, $\sum_{k=1}^{K} G_k = R$; here, we will take all G_k equal so that HA: $R = G_k \times K$ $G_k = G := R/K$. Thus, it is enough to specify $n, HA: G_k,$ and K. Table 1 summarizes various setups used in our simulations. We recall that p is the dimension of the feature vectors x_i (p) and "the noise level" is HA: equivalent to σ_r in (2). In each case, the number of groups R and the number of observations per group n_r is determined by the number of clusters G HA: In each case, the number of groups R and the number of observations per group n_r is determined by the number of clusters K, number of groups in each cluster G_k, and total number of observations n. For example, for $n = 800$, $G_k = 10$, and $K = 2$, we have $R = 20$ and $n_r = 40$. AA: This does not seem to be consistent with what you had before... HA: I made changes to be consistent with what we have done.

AA: Another idea for simulation: it would be interesting to somewhat vary R from 1 to n. It can't be 1, since G needs to be an integer... we can vary R as $K, 2K, \ldots, n$. The extreme case where there is only one observation per group, $n_r = 1$ or $R = n$, is interesting since it reduces to the usual mixture of regression. By doing this simulation, we can study the effect of having more observation per group than just one. HA: What n is intended to do is to vary the num. of observations per group (with fixed G_k and K, but the lowest is 10 observation per group while the highest is 40. I will proceed with this for now for my dissertation but will try to implement your suggestion before submitting the paper, enshaallah

To study the effect of heterogeneity among regression coefficient vectors $\beta_k, k \in [K]$, we take β_ks to be equi-distance points on the unit sphere in $\mathbb{R}^p$ and vary their common distance, denoted as δ_β. More precisely, we will have $\|\beta_k\| = 1$ HA: $\|\beta_k\| = \|\beta_\ell\|$ and $\|\beta_k - \beta_\ell\| = \delta_\beta$ for all $k \neq \ell$ for all $k \neq \ell$. AA: Is this really the case, did you get the norm to be 1 eventually? HA: No, they don't have norm 1 that is why I had to normalize the calculated error in each case Generating βs this way enables us to compare the estimation errors among different runs of the experiment. The comparison can be carried out across different setups by normalizing the calculated error, e.g. by the β-distance. Three values of the β distances sufficient for our purposes are selected (cf. Table 1).

Obviously, the smaller the distance, the closer the βs, and it is harder to separate the clusters.

Evaluation criteria. The Monte Carlo simulations are repeated 1000 times for each pair of β-distance and the noise level as well as pairs of d and K. Four criterion are used to benchmark the performance of the algorithm:

(1) Normalized mutual information (NMI) for assessing the clustering accuracy, (2) average β estimation error, (3) root mean squared error (RMSE) of prediction to assess the prediction power of the models, and (4) the number of iterations to study the rate of convergence and the speed of the algorithm.,

NMI is a widely used measure for evaluating quality of clustering algorithms when the true labels are available. Advantages of using NMI is that it is invariant to cluster label switching, and penalizes partitions close to random quite aggressively. NMI is bounded between zero and one. The closer the value to zero, the higher the indication that the cluster assignments are largely independent, while NMI close to one shows substantial agreement between the clusters.

"β estimation error" is used as another measure of goodness of fit. We calculate this error by considering both the distance between the true and estimated βs, as well as the miss-classification error. More precisely, to each group r, we can assign two regression coefficient vectors, the estimated one $\widehat{\beta}^{(r)}$, and the true one $\beta^{(r)}$; $\widehat{\beta}^{(r)}$ is equal to $\widehat{\beta}_k$ if we have estimated group r to be in cluster k. Similarly, $\beta^{(r)}$ is equal to β_k if group r is in true cluster k. We can define the average β estimation error as:

$$\text{avg err}_\beta := \frac{1}{R}\sum_{r=1}^{R} \|\widehat{\beta}^{(r)} - \beta^{(r)}\|^2 = \text{tr}(D^T F) \tag{9}$$

where $D = \left(\|\widehat{\beta}_k - \widehat{\beta}_\ell\|^2,\ k, \ell \in [K]\right)$ is the $K \times K$ matrix of pairwise squared distances between $\widehat{\beta}_k$s, and F is the confusion matrix between the estimated and true labels. The details for the second equality can be found in Appendix A.2. Prediction RMSE is obtained by designating a hold-out (or test) set and using the trained models to predict the responses over the hold-out set. AA: Maybe some detail about what fraction is HA: In each simulation run, 80% of the data is used in training, while 20 % is used as hold-out set.

3 Result

In this section a detailed discussion about the result of the simulation is presented. Each factor of the study is presented in a sub-section.

Table 2: NMI

		β dist. = 4					β dist. = 7					β dist. = 11				
	N Noise	**2**	**4**	**6**	**8**	**10**	**2**	**4**	**6**	**8**	**10**	**2**	**4**	**6**	**8**	**10**
K = 2; d = 2	**100**	0.88	0.43	0.22	0.13	0.09	0.99	0.87	0.62	0.43	0.27	0.99	0.98	0.88	0.71	0.54
	200	0.98	0.71	0.40	0.25	0.15	0.99	0.98	0.88	0.68	0.54	1	0.99	0.97	0.92	0.82
	400	0.99	0.91	0.68	0.46	0.32	1	0.99	0.98	0.90	0.81	1	1	0.99	0.98	0.96
	800	1	0.99	0.98	0.90	0.75	1	1	1	0.99	0.99	1	1	1	1	1
K = 2; d = 4	**100**	0.93	0.49	0.21	0.13	0.09	0.99	0.93	0.73	0.48	0.32	0.99	0.99	0.93	0.8	0.64
	200	0.99	0.82	0.45	0.26	0.16	1	0.99	0.94	0.80	0.62	1	0.99	0.99	0.97	0.91
	400	1	0.97	0.79	0.54	0.35	1	1	0.99	0.97	0.89	1	1	1	0.99	0.99
	800	1	0.99	0.96	0.84	0.64	1	1	1	0.99	0.98	1	1	1	1	0.99
K = 4; d = 4	**100**	0.80	0.34	0.20	0.15	0.12	0.97	0.80	0.52	0.34	0.25	0.98	0.94	0.81	0.62	0.45
	200	0.95	0.61	0.33	0.21	0.17	0.97	0.95	0.81	0.61	0.44	0.97	0.97	0.95	0.86	0.75
	400	0.96	0.86	0.58	0.38	0.27	0.96	0.96	0.94	0.86	0.72	0.96	0.96	0.96	0.95	0.92
	800	0.95	0.95	0.84	0.64	0.47	0.95	0.95	0.96	0.95	0.92	0.96	0.95	0.96	0.96	0.96

Table 3: β error

		β dist. = 4					β dist. = 7					β dist. = 11				
	N Noise	**2**	**4**	**6**	**8**	**10**	**2**	**4**	**6**	**8**	**10**	**2**	**4**	**6**	**8**	**10**
K = 2; d = 2	**100**	0.79	0.96	1.07	1.19	1.42	0.77	0.81	0.85	0.90	1.03	0.8	0.78	0.79	0.85	0.93
	200	0.76	0.81	0.97	1.06	1.15	0.78	0.81	0.79	0.87	0.90	0.79	0.76	0.78	0.81	0.81
	400	0.79	0.76	0.85	0.96	1.02	0.79	0.79	0.8	0.82	0.85	0.78	0.79	0.76	0.81	0.79
	800	0.07	0.1	0.14	0.21	0.31	0.06	0.07	0.08	0.1	0.12	0.06	0.06	0.07	0.08	0.09
K = 2; d = 4	**100**	0.21	0.59	0.95	1.19	1.41	0.11	0.59	0.38	0.59	0.78	0.09	0.14	0.21	0.32	0.45
	200	0.14	0.32	0.63	0.88	1.06	0.09	1.43	0.21	0.32	0.47	0.08	0.11	0.14	0.18	0.25
	400	0.11	0.19	0.35	0.55	0.76	0.08	0.10	0.14	0.19	0.26	0.07	0.09	0.11	0.13	0.16
	800	0.09	1.40	0.20	0.31	0.46	0.07	0.09	0.11	0.13	0.17	0.06	0.07	0.09	0.10	0.12
K = 4; d = 4	**100**	1.17	1.3	1.44	1.60	1.76	1.12	1.17	1.23	1.3	1.38	1.13	1.14	1.41	1.62	1.83
	200	1.14	1.23	1.31	1.40	1.51	1.14	1.15	1.17	1.20	1.26	1.13	1.12	1.15	1.16	1.19
	400	1.14	1.13	1.21	1.29	1.36	1.13	1.13	1.16	1.16	1.19	1.13	1.14	1.13	1.14	1.15
	800	1313	1.14	1.16	1.21	1.25	1.13	1.13	1.14	1.14	1.14	1.14	1.13	1.13	1.14	1.14

Table 4: RMSE

		β distance = 4					β distance = 7					β distance = 11				
	N Noise	**2**	**4**	**6**	**8**	**10**	**2**	**4**	**6**	**8**	**10**	**2**	**4**	**6**	**8**	**10**
K = 2; d = 2	**100**	1.49	2.12	2.87	3.71	4.54	1.23	1.48	1.77	2.10	2.49	1.2	1.30	1.47	1.67	1.88
	200	1.23	2.11	2.87	3.69	4.51	1.23	1.47	1.76	2.10	2.49	2.11	1.32	1.46	1.66	1.88
	400	1.46	2.11	2.88	3.68	4.52	1.25	1.48	1.78	2.13	2.48	2.98	1.32	1.46	1.66	1.89
	800	1.39	2.06	2.83	3.6	4.49	1.18	1.4	1.7	2.06	2.44	1.13	1.26	1.4	1.6	1.833
K = 2; d = 4	**100**	1.45	2.10	2.86	3.69	4.53	1.23	1.45	1.75	2.11	2.50	1.18	1.30	1.44	1.65	1.86
	200	1.45	2.10	2.87	3.68	4.51	1.23	1.45	1.75	2.10	2.48	1.19	1.30	1.45	1.64	1.86
	400	0.14	2.10	2.86	3.67	4.51	1.24	1.45	1.75	2.10	2.47	1.19	1.30	1.45	1.64	1.86
	800	1.44	2.09	2.86	3.67	4.50	1.23	1.45	1.74	2.09	2.46	1.19	1.29	1.45	1.65	1.86
K = 4; d = 4	**100**	1.41	2.07	2.85	3.67	4.50	1.18	1.41	1.72	2.07	2.45	1.14	1.25	1.41	1.62	1.83
	200	1.41	2.06	2.84	3.66	4.49	1.19	1.40	1.72	2.06	2.45	1.15	1.26	1.41	1.61	1.82
	400	1.41	2.07	2.84	3.66	4.50	1.19	1.41	1.72	2.06	2.44	1.14	1.25	1.41	1.60	1.83
	800	1.41	2.06	2.84	3.65	4.49	1.19	1.41	1.72	2.07	2.44	1.14	1.25	1.41	1.60	1.82

Table 5: Number of iterations

		β distance = 4					β distance = 7					β distance = 11				
	N Noise	**2**	**4**	**6**	**8**	**10**	**2**	**4**	**6**	**8**	**10**	**2**	**4**	**6**	**8**	**10**
K = 2; d = 2	**100**	14.1	53.7	84.2	100.2	109.0	4.9	14.3	32.9	53.7	71.3	3.8	7.5	13.6	27.2	40.1
	200	6.3	27.6	60.6	82.4	101.3	3.3	6.4	14.4	29.3	44.7	3.1	3.8	6.6	11.5	19.6
	400	3.6	12.3	33.2	55.4	74.1	2.9	3.6	6.4	12.9	20.4	2.8	3.0	3.5	5.6	8.3
	800	2.7	3.4	6.8	13.9	24.7	2.5	2.8	3	3.4	4.7	2.4	2.6	2.8	2.9	3.1
K = 2; d = 4	**100**	11.4	42.8	65.1	72.4	73.6	4.2	12.1	25.2	43.2	55.2	3.6	5.8	12.0	19.6	30.8
	200	4.8	20.6	45.8	65.4	73.4	3.2	4.8	11.2	20.5	32.8	3.1	3.4	4.7	8.2	13.7
	400	3.2	8.8	23.3	42.5	57.5	2.9	3.2	4.7	8.6	15.7	2.8	3.0	3.2	3.8	5.7
	800	2.8	4.0	9.5	19.7	34.2	2.5	2.8	3.1	3.8	6.1	2.4	2.6	2.8	3.1	3.3
K = 4; d = 4	**100**	41.1	114.0	149.7	163.2	171.1	13.0	40.5	81.1	113.4	134.1	8.5	20.1	39.3	67.9	94.1
	200	18.6	74.7	121.8	149.1	162.5	7.7	18.3	42.7	72.8	103.0	9.15	10.7	18.5	33.2	52.3
	400	11.6	36.5	81.5	117.1	142.7	11.3	12.3	20.2	36.2	59.8	11.0	11.8	12.8	17.1	25.4
	800	12.7	18.1	40.5	72.8	104.6	13.0	13.2	12.0	17.8	27.7	10.1	12.6	12.7	12.5	14.0

3.1 Dimension (d)

4 Case Study: Deriving Recommendations for Automotive Dealerships

4.1 Applying the Algorithm to the Dealership Performance Problem

4.1.1 Results

5 Conclusion

Appendix A Derivations

A.1 The EM updates in Algorithm 1

Expanding the expected log-likelihood (8) using the definition of $\gamma_{rk}(\theta)$ in (4), we have

$$F(\theta;\widehat{\theta}) = E_{z\sim\tau(\widehat{\theta})}[\ell(\theta;z)] = \sum_{k=1}^{K}\tau_{+k}(\widehat{\theta})\log\pi_k + \sum_{r=1}^{R}\sum_{k=1}^{K}\sum_{i=1}^{n_r}\tau_{rk}(\widehat{\theta})\log\phi_{\sigma_r}\big(y_{ri}-\beta_k^T x_{ri}\big). \quad (10)$$

where $\phi_\sigma(t) := (2\pi\sigma^2)^{-1/2}\exp(-\frac{1}{2}t^2/\sigma^2)$ is the density of $N(0,\sigma^2)$.

We would like to maximize (11) over θ. Recall that $\beta_k, x_{ri} \in \mathbb{R}^p$ where p is the number of features. We will use $\doteq_\pi$ for example, when the two sides are equal up to additive constants, as functions of π. Fixing everything and maximizing over $\pi = (\pi_1,\ldots,\pi_k)$, we are maximizing $\pi \mapsto \sum_k \tau_{+k}(\widehat{\theta})\log\pi_k$ over probability vector π. This is the MLE in the multinomial family and the solution is $\pi_k \propto_k \tau_{+k}$, that is

$$\pi_k = \frac{\tau_{+k}}{\sum_{k'}\tau_{+k'}} = \frac{\tau_{+k}}{R} \quad (11)$$

where we used $\sum_{k'}\tau_{+k'} = \sum_{k'}\sum_r \tau_{rk'} = \sum_r\sum_{k'}\tau_{rk'} = \sum_r 1 = R$, since for fixed r, τ_{rk} sums to 1 over k.

To maximize over β, we again fix everything else. Since $\log\phi_\sigma(t) \doteq_t -\frac{1}{2}(\log\sigma^2 + t^2/\sigma^2)$, we are maximizing

$$\begin{aligned} F(\theta;\widehat{\theta}) &\doteq_\beta -\sum_r\sum_k\sum_i^{n_r}\tau_{rk}(\widehat{\theta})\frac{1}{2\sigma_k^2}(y_{ri}-\beta_k^T x_{ri})^2 \\ &\doteq_\beta -\sum_r\sum_k\sum_i^{n_r}\tau_{rk}(\widehat{\theta})\frac{1}{2\sigma_k^2}[(\beta_k^T x_{ri})^2 - 2y_{ri}\beta_k^T x_{ri}] \end{aligned} \quad (12)$$

ignoring the constant terms generated by y_{ri}^2. Note that $(\beta_k^T x_{ri})^2 = (\beta_k^T x_{ri})(x_{ri}^T\beta_k) = \beta_k^T(x_{ri}x_{ri}^T)\beta_k$. Similarly, $y_{ri}\beta_k^T x_{ri} = \beta_k^T(y_{ri}x_{ri})$. Let us define

$$\widehat{\Sigma}_r := \frac{1}{n_r}\sum_{i=1}^{n_r} x_{ri}x_{ri}^T, \quad \widehat{\rho}_r := \frac{1}{n_r}\sum_{i=1}^{n_r} y_{ri}x_{ri} \quad (13)$$

Summing over i first in (13), we get

$$F(\theta;\widehat{\theta}) \doteq_\beta -\sum_r \sum_k \frac{\tau_{rk}}{2\sigma_k^2} n_r [\beta_k^T \widehat{\Sigma}_r \beta_k - 2\beta_k^T \widehat{\rho}_r] = -\sum_k \frac{1}{2\sigma_k^2} \sum_r \tau_{rk} n_r [\beta_k^T \widehat{\Sigma}_r \beta_k - 2\beta_k^T \widehat{\rho}_r] \tag{14}$$

Let us define $w_{rk} := n_r \tau_{rk}$ and $\check{w}_{rk} := w_{rk}/w_{+k}$ where $w_{+k} = \sum_r n_r \tau_{rk}$, and let

$$\widetilde{\Sigma}_k := \sum_{r=1}^R \check{w}_{rk} \widehat{\Sigma}_r, \quad \widetilde{\rho}_k := \sum_{r=1}^R \check{w}_{rk} \widehat{\rho}_r. \tag{15}$$

Summing over i first in (13), we get

$$F(\theta;\widehat{\theta}) \doteq_\beta -\sum_r \sum_k \frac{\tau_{rk}}{2\sigma_k^2} n_r [\beta_k^T \widehat{\Sigma}_r \beta_k - 2\beta_k^T \widehat{\rho}_r] = -\sum_k \frac{1}{2\sigma_k^2} \sum_r \tau_{rk} n_r [\beta_k^T \widehat{\Sigma}_r \beta_k - 2\beta_k^T \widehat{\rho}_r] \tag{16}$$

Let us define $w_{rk} := n_r \tau_{rk}$ and $\check{w}_{rk} := w_{rk}/w_{+k}$ where $w_{+k} = \sum_r n_r \tau_{rk}$, and let

$$\widetilde{\Sigma}_k := \sum_{r=1}^R \check{w}_{rk} \widehat{\Sigma}_r, \quad \widetilde{\rho}_k := \sum_{r=1}^R \check{w}_{rk} \widehat{\rho}_r. \tag{17}$$

Dividing and multiplying by w_{+k} and summing over r in (17), we get $F(\theta;\widehat{\theta}) \doteq_\beta -\sum_k \frac{w_{+k}}{2\sigma_k^2} [\beta_k^T \widetilde{\Sigma}_k \beta_k - 2\beta_k^T \widetilde{\rho}_k]$. The problem is separable in k, and the minimizer over β_k is $\beta_k = \widetilde{\Sigma}_k^{-1} \widetilde{\rho}_k$.

To optimize over $\alpha_k := \sigma_k^2$, let us fix everything else. We have

$$F(\theta;\widehat{\theta}) \doteq_\alpha -\frac{1}{2} \sum_k \Big[\sum_r \sum_i^{n_r} \tau_{rk} \log \alpha_k + \sum_r \sum_i^{n_r} \tau_{rk} \frac{(y_{ri} - \beta_k^T x_{ri})^2}{\alpha_k} \Big]. \tag{18}$$

The first term in brackets is $(\sum_r n_r \tau_{rk}) \log \alpha_k = w_{+k} \log \alpha_k$. Defining

$$E_{rk} := E_{rk}(\beta) := \frac{1}{n_r} \sum_i^{n_r} (y_{ri} - \beta_k^T x_{ri})^2, \qquad \overline{E}_k := \overline{E}_k(\beta) := \sum_r \check{w}_{rk} E_{rk}. \tag{19}$$

we see that the second term in brackets in (19) is just $w_{+k}\overline{E}_k$. We have

$$F(\theta;\widehat{\theta}) \doteq_\alpha -\frac{1}{2}\sum_k w_{+k}\Big[\log\alpha_k + \frac{\overline{E}_k}{\alpha_k}\Big] \tag{20}$$

This problem is separable in α_k and the solution is $\alpha_k = \overline{E}_k$. Putting the pieces together, we obtain the Algorithm 1.

A.2 Details of (10)

Let $\widehat{C}_k \subset [R]$ be the kth estimated cluster (the set containing indices of the groups estimated to be in cluster k) and $\widehat{z}_r \in \{0,1\}^K$ the estimated membership vector for group r, so that $\widehat{z}_{rk} = 1\{r \in \widehat{C}_k\}$. Similarly, let $C_k \subset [R]$ be the true cluster k and z_r the true label vector for group r, so that $z_{rk} = 1\{z_r \in C_k\}$. The normalized confusion matrix $F = (F_{k\ell}) \in [0,1]^{K\times K}$ between the two sets of labels is given by $F_{k\ell} = \frac{1}{R}\sum_{r=1}^R z_{rk}\,\widehat{z}_{r\ell} = \frac{1}{R}\sum_{r=1} 1\{r \in C_k,\, r \in \widehat{C}_\ell\}$.

References

Andrews, R. L. and I. S. Currim (2003). Retention of latent segments in regression-based marketing models. *International Journal of Research in Marketing 20*(4), 315–321.

Bar-Shalom, Y. (1978). Tracking methods in a multitarget environment. *Automatic Control, IEEE Transactions on 23*(4), 618–626.

Basu, S. (2009). *Constrained clustering : Advances in algorithms, theory, and applications*. Boca Raton: CRC Press.

Berkhin, P. (2006). *A Survey of Clustering Data Mining Techniques*, pp. 25–71. Berlin, Heidelberg: Springer Berlin Heidelberg.

Bierbrauer, M., S. Trück, and R. Weron (2004). Modeling electricity prices with regime switching models. In *Computational Science-ICCS 2004*, pp. 859–867. Springer.

Böhning, D. (2000). *Computer-assisted analysis of mixtures and applications: meta-analysis, disease mapping and others*, Volume 81. CRC press.

Celeux, G. and J. Diebolt (1985). The sem algorithm: a probabilistic teacher algorithm derived from the em algorithm for the mixture problem. *Computational statistics quarterly 2*(1), 73–82.

Celeux, G. and G. Govaert (1992). A classification em algorithm for clustering and two stochastic versions. *Computational statistics & Data analysis 14*(3), 315–332.

Chaganty, A. T. and P. Liang (2013). Spectral experts for estimating mixtures of linear regressions. *arXiv preprint arXiv:1306.3729*.

Davidson, I. and S. Basu (2007). A survey of clustering with instance level constraints. *ACM Transactions on Knowledge Discovery from Data*, 1–41.

Davidson, I. and S. Ravi (2005). Agglomerative hierarchical clustering with constraints: Theoretical and empirical results. In *European Conference on Principles of Data Mining and Knowledge Discovery*, pp. 59–70. Springer.

Davidson, I., S. Ravi, and M. Ester (2007). Efficient incremental constrained clustering. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 240–249. ACM.

De Veaux, R. D. (1989). Mixtures of linear regressions. *Computational Statistics & Data Analysis 8*(3), 227–245.

Dempster, A. P., N. M. Laird, and D. B. Rubin (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society. Series B (methodological)*, 1–38.

Dhillon, I. S., J. Fan, and Y. Guan (2001). Efficient clustering of very large document collections. *Data mining for scientific and engineering applications 2*, 357–381.

Diebolt, J. and C. P. Robert (1994). Estimation of finite mixture distributions through bayesian sampling. *Journal of the Royal Statistical Society. Series B (Methodological)*, 363–375.

Faria, S. and G. Soromenho (2010). Fitting mixtures of linear regressions. *Journal of Statistical Computation and Simulation 80*(2), 201–225.

Hurn, M., A. Justel, and C. P. Robert (2003). Estimating mixtures of regressions. *Journal of Computational and Graphical Statistics 12*(1), 55–79.

Law, M. H., A. Topchy, and A. K. Jain (2005). Model-based clustering with probabilistic constraints. In *Proceedings of the 2005 SIAM international conference on data mining*, pp. 641–645. SIAM.

Lemke, W. (2006). *Term structure modeling and estimation in a state space framework*, Volume 565. Springer Science & Business Media.

McLachlan, G. and D. Peel (2004). *Finite mixture models*. John Wiley & Sons.

MORTIER, F., D. OU, and N. PICARD. Finite mixture of regression model and adaptive lasso selection approaches to predict growth, mortality and recruitment of tropical tree species.

Newcomb, S. (1886). A generalized theory of the combination of observations so as to obtain the best result. *American Journal of Mathematics*, 343–366.

Pearson, K. (1894). Contributions to the mathematical theory of evolution. *Philosophical Transactions of the Royal Society of London. A 185*, 71–110.

Permuter, H., J. Francos, et al. (2003). Gaussian mixture models of texture and colour for image database retrieval. In *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*, Volume 3, pp. III–569. IEEE.

Quandt, R. E. and J. B. Ramsey (1978). Estimating mixtures of normal distributions and switching regressions. *Journal of the American statistical Association 73*(364), 730–738.

Reynolds, D., R. C. Rose, et al. (1995). Robust text-independent speaker identification using gaussian mixture speaker models. *Speech and Audio Processing, IEEE Transactions on 3*(1), 72–83.

Segal, E., H. Wang, and D. Koller (2003). Discovering molecular pathways from protein interaction and gene expression data. *Bioinformatics 19*(suppl 1), i264–i272.

Stylianou, Y., Y. Pantazis, F. Calderero, P. Larroy, F. Severin, S. Schimke, R. Bonal, F. Matta, and A. Valsamakis (2005). Gmm-based multimodal biometric verification. In *eNTERFACE 2005 The summer Workshop on Multimodal Interfaces July 18th–August 12th, Faculte Polytechnique de Mons, Belgium*.

Tuma, M. and R. Decker (2013). Finite mixture models in market segmentation: a review and suggestions for best practices. *Electronic Journal of Business Research Methods 11*(1).

Wagstaff, K., C. Cardie, S. Rogers, and S. Schrödl (2001). Constrained k-means clustering with background knowledge. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, San Francisco, CA, USA, pp. 577–584. Morgan Kaufmann Publishers Inc.

Wagstaff, K. L. (2002). *Intelligent clustering with instance-level constraints*. Ph. D. thesis, Cornell University.

Xing, E. P., A. Y. Ng, M. I. Jordan, and S. Russell (2002). Distance metric learning with application to clustering with side-information. In *NIPS*, Volume 15, pp. 12.

Yan, R., J. Zhang, J. Yang, and A. G. Hauptmann (2006). A discriminative learning framework with pairwise constraints for video object classification. *IEEE transactions on pattern analysis and machine intelligence 28*(4), 578–593.

Yang, K.-S., R. Yang, and M. Kafatos (2001). A feasible method to find areas with constraints using hierarchical depth-first clustering. In *Scientific and Statistical Database Management, 2001. SSDBM 2001. Proceedings. Thirteenth International Conference on*, pp. 257–262. IEEE.