Generative Linear Mixture Models

Antony Lawson and Jochen Einbeck

Department of Mathematical Sciences, Durham University

Introduction

Mixtures of exponential family distributions are often used to model complex data structures, with finite Gaussian mixtures being the most common representant of such models. We are interested in situations where a multivariate data set, $\boldsymbol{x_i} \in \mathbb{R}^m$, $i = 1, \dots, n$, possesses a latent structure of lower dimension $d < m$ (these 'data' may play the role of a 'predictor space' in a multivariate regression problem, but this is not relevant for the moment). The objective, for now, is to recover the latent structure, and to compress the original data by projecting them (in some form) onto the estimated latent space. As a first step towards a more general handling of this problem, we consider a simplified scenario in which the latent structure is thought to be a straight line, say $\alpha + \beta z$, with $\alpha, \beta \in \mathbb{R}^m$, $z \in \mathbb{R}$, through an m-dimensional space. The parameter z is considered as a random effect, and represented by a discrete distribution with mass points $z_k \in \mathbb{R}$ and masses $\pi_k, k = 1, \dots, K$. The data are assumed to be generated by adding Gaussian noise $\varepsilon_i \sim N(0, \Sigma)$ to mixture centres $\alpha + \beta z_k \in \mathbb{R}^m$ positioned along this line, yielding the generative linear mixture model (GLMM)

$$x_i = \alpha + \beta z_k + \varepsilon_i. \tag{1}$$

The variance matrix $\Sigma \in \mathbb{R}^{m \times m}$ is assumed to be of diagonal form $\text{diag}(\sigma_j^2)_{\{1 \le j \le m\}}$, and to be the same for all K components of the mixture

The EM Algorithm

As for a mixed exponential family distribution, the GLMM we propose in (1) is fitted using an EM algorithm. The EM algorithm is an iterative procedure used to acquire maximum likelihood estimates of model parameters.
As for univariate mixtures, the data likelihood, L, can be written in the form

$$L = \prod_{i=1}^n \sum_{k=1}^K f_{ik}\pi_k$$

where, for model (1),

$$f_{ik} = f(x_i|z_k) = \frac{1}{|\Sigma|^{\frac{1}{2}}} \frac{1}{(2\pi)^{m/2}} \exp\left(-\frac{1}{2}(x_i - \alpha - \beta z_k)^T \Sigma^{-1} (x_i - \alpha - \beta z_k)\right)$$

In order to setup an EM algorithm, we need to consider the complete data likelihood, which is the likelihood of the data given that we know the component each x_i belongs to. However, the components each datum belongs to are unobservable, so we must use the posterior probabilities that x_i belongs to component k, which are obtained as

$$\omega_{ik} = \frac{f_{ik}\pi_k}{\sum_{l=1}^K f_{il}\pi_l}.$$

The complete data likelihood therefore takes the form

$$L^* = \prod_{i=1}^n \prod_{k=1}^K (f_{ik}\pi_k)^{\omega_{ik}},$$

giving the complete log-likelihood

$$\ell^* = \log(L^*) = \sum_{i=1}^n \sum_{k=1}^K \omega_{ik} \log(\pi_k) + \sum_{i=1}^n \sum_{k=1}^K -\frac{1}{2}\omega_{ik} \log(|\Sigma|)$$
$$+ \sum_{i=1}^n \sum_{k=1}^K -\omega_{ik}\frac{m}{2}\log(2\pi) + \sum_{i=1}^n \sum_{k=1}^K -\frac{1}{2}\omega_{ik}(x_i - \alpha - \beta z_k)^T \Sigma^{-1}(x_i - \alpha - \beta z_k)$$

When using the EM algorithm, some initialization of values is used. Then, in the M-step the model parameters (here the z_k, σ_j, α, β and π_k) are estimated using the ω_{ik}. Then, in the E-step the ω_{ik} are updated using the parameters estimated in the previous M-step. E-steps and M-steps are looped iteratively until some convergence criterion is met.

Extending the Methodology to Planes

Given the success of our GLMM for modelling data whose latent structure is a straight line, it seemed natural to extend the methodology to data whose latent structure is 2-dimensional (i.e. a plane). This is achieved by now assuming the mixture centres to be $\alpha + \beta z_k + \gamma u_k \in \mathbb{R}^m$, with $\alpha, \beta, \gamma \in \mathbb{R}^m$; $z_k, u_k \in \mathbb{R}$. The GLMM now has the form

$$x_i = \alpha + \beta z_k + \gamma u_k + \varepsilon_i. \tag{2}$$

We again assume the variance matrix to be diagonal and the same for each component.
The resultant complete data log-likelihood equation is

$$\ell^* = \sum_{i=1}^n \sum_{k=1}^K \omega_{ik} \log(\pi_k) + \sum_{i=1}^n \sum_{k=1}^K -\frac{1}{2}\omega_{ik} \log(|\Sigma|) + \sum_{i=1}^n \sum_{k=1}^K -\omega_{ik}\frac{m}{2}\log(2\pi)$$
$$+ \sum_{i=1}^n \sum_{k=1}^K -\frac{1}{2}\omega_{ik}(x_i - \alpha - \beta z_k - \gamma u_k)^T \Sigma^{-1}(x_i - \alpha - \beta z_k - \gamma u_k)$$

Again, we were unable to find an analytical solution to the resulting score equations, but an iterative procedure could be implemented using the following results:

- $\hat{z}_k = \sum_{j=1}^m \left(\frac{\sum_{i=1}^n \omega_{ik} x_i}{\sum_{i=1}^n \omega_{ik}} - \hat{\alpha}_j - \hat{\gamma}_j \hat{u}_k\right) / \sum_{j=1}^m \hat{\beta}_j$
- $\hat{u}_k = \sum_{j=1}^m \left(\frac{\sum_{i=1}^n \omega_{ik} x_i}{\sum_{i=1}^n \omega_{ik}} - \hat{\alpha}_j - \hat{\beta}_j \hat{z}_k\right) / \sum_{j=1}^m \hat{\gamma}_j$
- $\hat{\alpha} = \frac{1}{n}\left(\sum_{i=1}^n x_i - \hat{\beta}\sum_{i=1}^n \sum_{k=1}^K \omega_{ik}\hat{z}_k - \hat{\gamma}\sum_{i=1}^n \sum_{k=1}^K \omega_{ik}\hat{u}_k\right)$
- $\hat{\beta} = \dfrac{\sum_{i=1}^n \sum_{k=1}^K \omega_{ik} x_i \hat{z}_k - \hat{\gamma}\sum_{i=1}^n \sum_{k=1}^K \omega_{ik}\hat{u}_k\hat{z}_k - \frac{1}{n}\left(\sum_{i=1}^n x_i\right)\left(\sum_{i=1}^n \sum_{k=1}^K \omega_{ik}\hat{z}_k\right) + \frac{\hat{\gamma}}{n}\left(\sum_{i=1}^n \sum_{k=1}^K \omega_{ik}\hat{z}_k\right)\left(\sum_{i=1}^n \sum_{k=1}^K \omega_{ik}\hat{u}_k\right)}{\sum_{i=1}^n \sum_{k=1}^K \omega_{ik}\hat{z}_k^2 - \frac{1}{n}\left(\sum_{i=1}^n \sum_{k=1}^K \omega_{ik}\hat{z}_k\right)^2}$
- $\hat{\gamma} = \dfrac{\sum_{i=1}^n \sum_{k=1}^K \omega_{ik} x_i \hat{u}_k - \hat{\beta}\sum_{i=1}^n \sum_{k=1}^K \omega_{ik}\hat{u}_k\hat{z}_k - \frac{1}{n}\left(\sum_{i=1}^n x_i\right)\left(\sum_{i=1}^n \sum_{k=1}^K \omega_{ik}\hat{u}_k\right) + \frac{\hat{\beta}}{n}\left(\sum_{i=1}^n \sum_{k=1}^K \omega_{ik}\hat{z}_k\right)\left(\sum_{i=1}^n \sum_{k=1}^K \omega_{ik}\hat{u}_k\right)}{\sum_{i=1}^n \sum_{k=1}^K \omega_{ik}\hat{u}_k^2 - \frac{1}{n}\left(\sum_{i=1}^n \sum_{k=1}^K \omega_{ik}\hat{u}_k\right)^2}$
- $\hat{\sigma}_j = \sqrt{\frac{1}{n}\sum_{i=1}^n \sum_{k=1}^K \omega_{ij}(y_{ij} - \hat{\alpha}_j - \hat{\beta}_j\hat{z}_k - \hat{\gamma}_j\hat{u}_k)^2}$
- $\hat{\pi}_k = \frac{\sum_{i=1}^n \omega_{ik}}{n}$

It was found using a loop which solved in the variables in the order $\hat{z}_k, \hat{\beta}, \hat{u}_k, \hat{\gamma}, \hat{\alpha}$ left the $\hat{u}_k$ unchanged (possibly because $\hat{z}_k$ and $\hat{u}_k$ use essentially the same score equation). This largely restricted the mass points, so the EM algorithm was changed such that one EM loop would use an M-step which solves in the order $\hat{z}_k, \hat{\beta}, \hat{u}_k, \hat{\gamma}, \hat{\alpha}$ and the next used an M-step which solves in the order $\hat{u}_k, \hat{\gamma}, \hat{z}_k, \hat{\beta}, \hat{\alpha}$. This vastly improved results and allowed us to give the data a much better fit.
We will consider in detail the example of fitting the GLMM of Equation (2) to the airquality dataset in the R package **datasets**, using Solar.R, Wind and Temp as 3 predictor covariates.

Implementing the EM Algorithm

Score equations were obtained by partially differentiating ℓ^* with respect to each of the variables. Although an analytical solution was not obtained for α, β and z_k, we were able to find an iteration process involving the three variables.
Solving the score equations for α and z_k give

$$\hat{z}_k = \frac{1}{m}\sum_{j=1}^m \left(\frac{\sum_{i=1}^n \omega_{ik} x_{ij}}{\sum_{i=1}^n \omega_{ik}} - \hat{\alpha}_j\right) / \hat{\beta}_j$$

(with the subscript j denoting the $j-$th component of the respective vector), and

$$\hat{\alpha} = \frac{1}{n}(\sum_{i=1}^n x_i - \hat{\beta}\sum_{i=1}^n \sum_{k=1}^K \omega_{ik}\hat{z}_k).$$

Substituting $\hat{\alpha}$ into the equation for $\hat{\beta}$ and solving gives

$$\hat{\beta} = \frac{\sum_{i=1}^n \sum_{k=1}^K \omega_{ik} x_i \hat{z}_k - \frac{1}{n}\left(\sum_{i=1}^n x_i\right)\left(\sum_{i=1}^n \sum_{k=1}^K \omega_{ik}\hat{z}_k\right)}{\sum_{i=1}^n \sum_{k=1}^K \omega_{ik}\hat{z}_k^2 - \frac{1}{n}\left(\sum_{i=1}^n \sum_{k=1}^K \omega_{ik}\hat{z}_k\right)^2}.$$

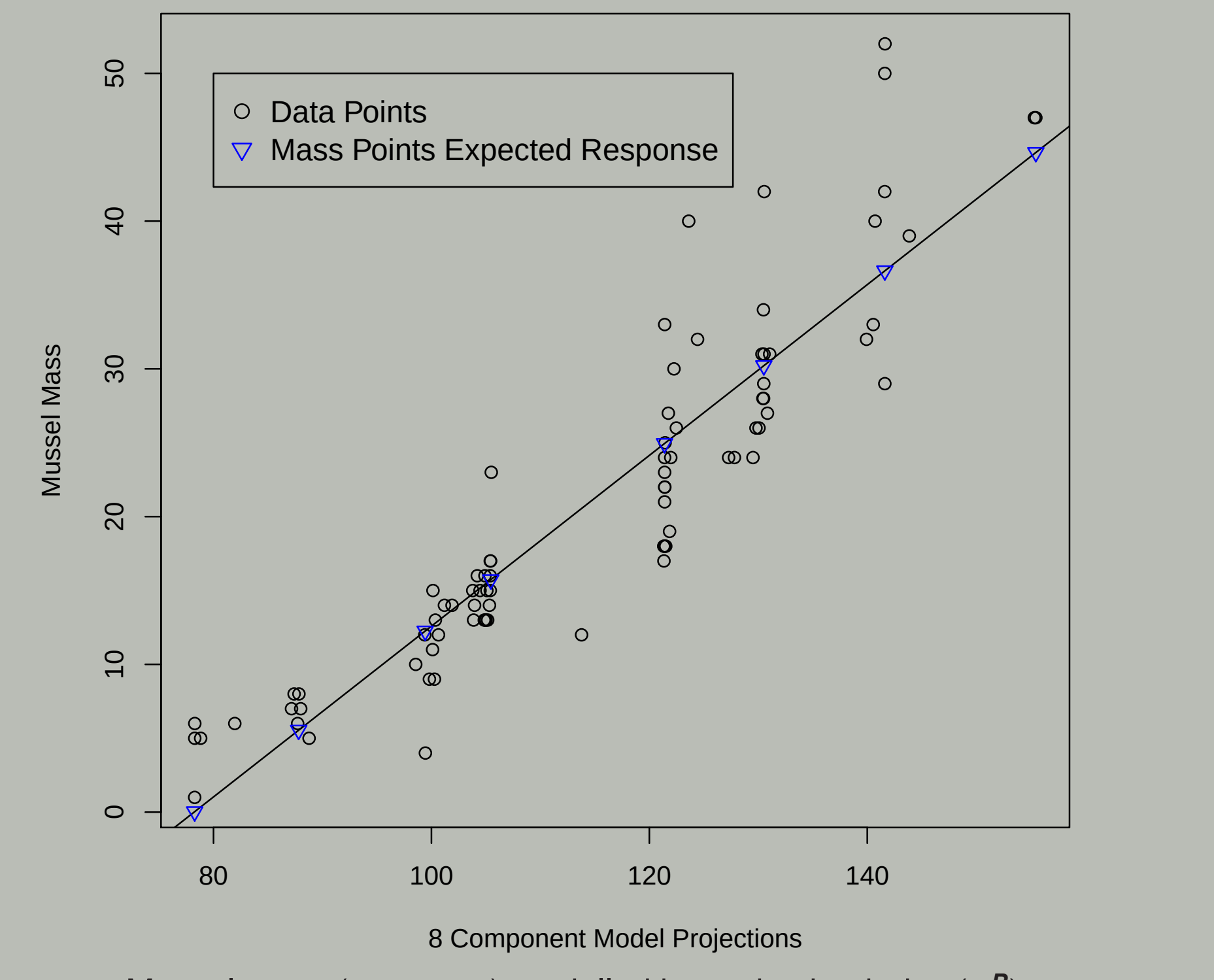
To implement this in the EM algorithm, at each M-step there will be an internal iteration loop involving these variables. First, the $\hat{z}_k$ will be calculated using the values of the previous internal iteration. Then $\hat{\beta}$ will be calculated using the newly calculated values of $\hat{z}_k$. Then finally $\hat{\alpha}$ will be calculated using the new values of $\hat{\beta}$ and $\hat{z}_k$. The initial $\hat{\beta}$ and $\hat{\alpha}$ values used will be those from the previous E-step. Given the new values of $\hat{\alpha}, \hat{\beta}$ and $\hat{z}_k$, the score equation for $\hat{\sigma}_j$ solves very easily to

$$\hat{\sigma}_j = \sqrt{\frac{1}{n}\sum_{i=1}^n \sum_{k=1}^K \omega_{ij}(x_{ij} - \hat{\alpha}_j - \hat{\beta}_j\hat{z}_k)^2}$$

Using a Lagrange multiplier, $\ell^* - \lambda(\sum_{k=1}^K \pi_k - 1)$, one obtains

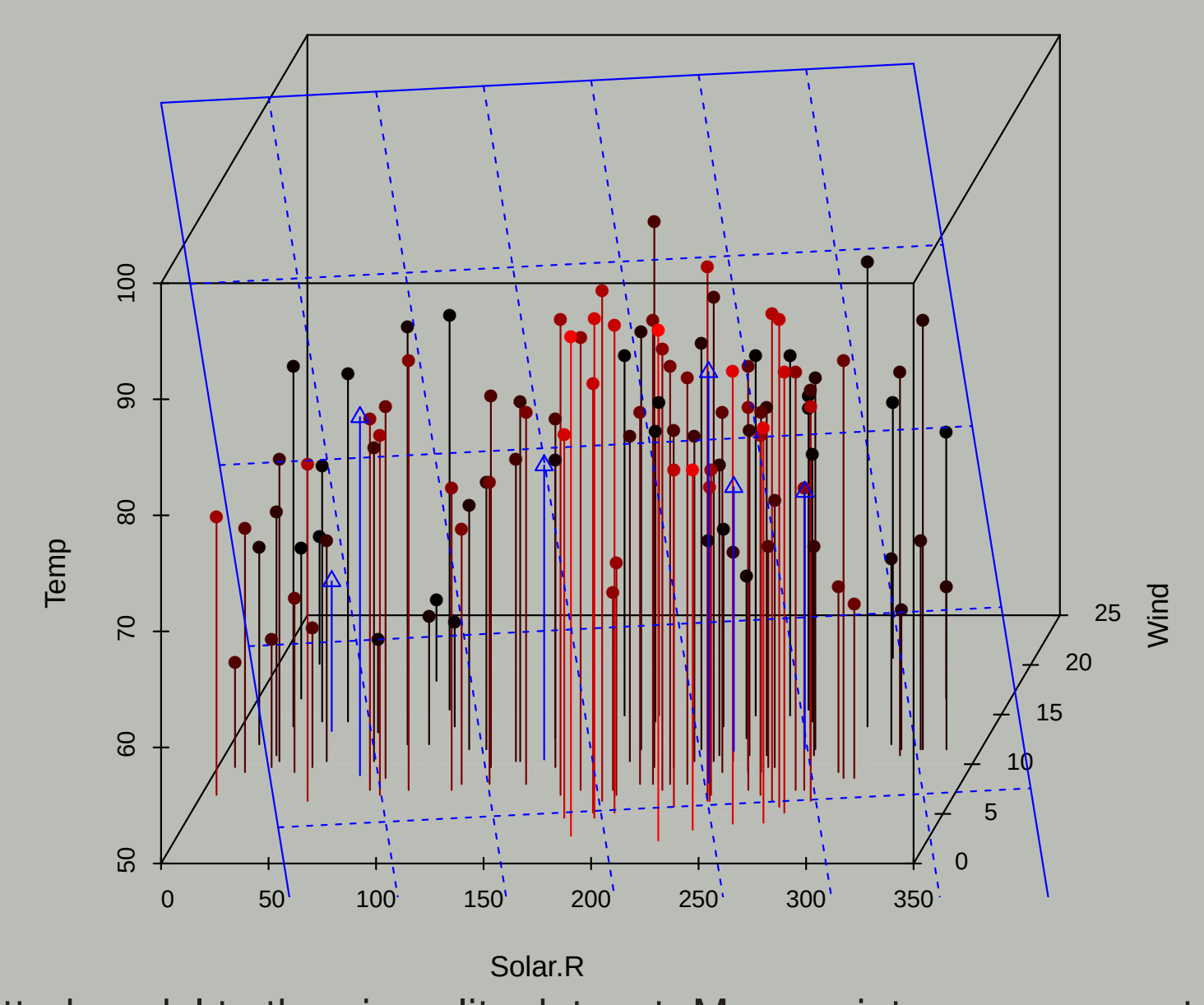
$$\hat{\pi}_k = \frac{1}{n}\sum_{i=1}^n \omega_{ik}.$$

Figure 1 : Application of Our GLMM



Mussel mass (response) modelled by projection index (x_i^P).

Figure 2 : A 2-Dimensional GLMM



Graph of our fitted model to the airquality dataset. Mass points are represented by blue triangles and the plane spanned by these mass points is also shown.

Selecting Appropriate Starting Values

Selecting appropriating starting values was a rather troublesome process. Initially a grid of mass points was used, but the resulting algorithm was very slow to converge and the mass points tended to cluster. Some alternative methods were proposed and we will concentrate on explaining what we called the 'k-means method'. Here, a k-means sample was taken of the data space, then the starting values set $\hat{z}_k$ to values of the k-means sample corresponding to the Solar.R and $\hat{u}_k$ to values of the k-means sample corresponding to the Wind. A linear model was fitted modelling the values of the k-means sample corresponding to Temp value by the values of the k-means sample corresponding to Solar.R and Wind values and this initialized $\hat{\alpha}$ to (0,0,intercept of linear model), $\hat{\beta}$ to (1,0,gradient associated with Solar.R), $\hat{\gamma}$ to (0,1,gradient associated with Wind).
A small problem with this method is results were not entirely consistent and the mass points and resulting disparity may vary slightly each time the algorithm is run. This effect is, however, very small when the data is truly clustered.
Another idea considered was to use the plane spanned by the first two principal components as a starting plane, and this method produced very similar results to the k-means method.

Table 1

Table of measurements for a variety of components

K	Disparity	RSS	R^2	# Iterations for disparity to converge
2	2881.936	7.057	0.6421	7
3	2804.671	5.574	0.7767	26
4	2676.017	5.222	0.8041	13
5	2645.342	5.073	0.8151	34
6	2630.438	5.010	0.8196	102
7	2623.526	4.783	0.8356	126
8	2608.088	4.759	0.8373	145
9	2608.088	4.759	0.8373	194

Results 1

Analysis was carried out on the mussels data set (Bura and Cook, 2001; available from R package **dr**), considering intially the data frame constituted by the variables shell length, shell width, shell height, and shell mass. Applying the above methodology, Table 1 shows how the disparity, $-2\log L$, of the model varies with number of components, K. The disparity decreases considerably with each component added, until the 8th component, where the disparity stabilizes at a value of 2608.088.
A bootstrapping method was required to test for a sensible number of components. Testing a model with 5 components against one with 6 returned a p-value of 0.31, and testing 4 components against 5 returned a p-value of 0.01, implying a 5 component model is a good representation of the data.
The iteration loop in the M-step converges very fast, with not more than 5 iterations initially, quickly falling to 3 iterations after a few EM cycles. The number of EM iterations taken for the variables to converge was also observed and the $\hat{\sigma}_j$ were generally the fastest to converge, with $\hat{\beta}$ converging slower, $\hat{\alpha}$ and $\hat{z}_k$ a little slower than $\hat{\beta}$, and $\hat{\pi}_k$ considerably slower. The disparity of the models converge somewhat faster than any of the components.
The next step taken in the analysis was projecting the data points onto the fitted line. For each data point x_i, projected (or compressed) data are obtained as 'posterior means' (Aitkin, 1996)

$$x_i^P = \sum_{k=1}^K \omega_{ik}\hat{z}_k.$$

These 'projections' are not orthogonal, and hence are of fundamentally different character as those in PCA, for instance. To verify the usefulness of this type of compression, we considered now the additional variable mussel mass as response variable, y, and fitted a simple linear regression model for y_i versus x_i^P. The resultant line is shown along with $(x_i^P, y_i), i = 1, \dots, n$ in Figure 1 and appears to represent the data reasonably well. The RSS and R^2 values for the fitted linear model were recorded for each model and are included in Table 1. It is clear that the model improves as number of components is increased. Comparing these results to the 'parametric inverse regression' method by Bura and Cook (2001), with RSS $= \mathbf{6.051}$ and $R^2 = \mathbf{0.741}$, we find the proposed mixture–based approach to perform considerably better.

Results 2

Figure 2 shows a fitted GLMM to the air quality dataset using 6 mass points. The plane seems to be a good fit of the data, and the mass points seem to identify the clustering of the data well. Bootstrapping methods were again necessary to test for a reasonable number of components, and 6 mass points seems to be an appropriate choice.

Discussion

We have conducted a pilot study using simple latent model scenarios, namely a straight line and a plane spanned through the data which carry the mixture centres. This research has also been tentatively extended in the case of a 'staggered line' which is an extension of the straight line scenario where the line allowed to change its direction at each mass point. The likelihood equation was still tractable and the resultant algorithm converged in reasonable time with very promising results, as the resulting 'curve' allowed for a more accurate fits to data.
The presented work could be considered as a generalization of the (linear version of the) 'Generative topographic mapping' (Bishop et al, 1998), where the z_k form a fixed grid, and $\pi_k = 1/K$. Using a grid to capture the latent variable distribution may require a quite large value of K, especially when considering multivariate latent structures. Since our method recovers adaptively the latent variable distribution, the value K can be kept on a far smaller level (say, 6 or 7) even for a bivariate latent space (i.e., a plane).
A further interesting aspect of the proposed technique is that, due to the generative model structure, it would allow additionally for inclusion of covariates in model (1). This would be attractive, for instance, for the computation of league tables from multivariate index data. This is a matter of further research.

Acknowledgements

The first author was supported by Nuffield Bursary URB/39353.00.
The theme used to create this poster is heavily based on the theme Dreuw.

References

[Aitkin, M.] (1996) Empirical Bayes shrinkage using posterior random ef- fect means from nonparametric maximum likelihood estimation in general random effect models. In: *Proceedings of the 11th IWSM 1996, 87–94, Orvieto, Italy.*
[Bishop, C.M J., Svensen, M. and Williams, C.K.I](1998) The Generative Topographic Mapping. *Neural Computation*, ***10***, 215–234.
[Bura, E. and Cook, R.D.](2001) Estimating the structural dimension of regressions via parametric inverse regression. *Journal of the Royal Statistical Society, Series B*, **63, 393–410**