Gene expression

Clustering of temporal gene expression data with mixtures of mixed effects models with a penalized likelihood

Darlene Lu^{1,*}, Yorghos Tripodis¹, Louis C. Gerstenfeld² and Serkalem Demissie¹

¹Biostatistics and ²Orthopaedic Surgery, Boston University, Boston, MA 02118, USA

*To whom correspondence should be addressed.

Associate Editor: Jonathan Wren

Received on April 12, 2018; revised on July 16, 2018; editorial decision on August 6, 2018; accepted on August 7, 2018

Abstract

Motivation: Clustering algorithms like K-Means and standard Gaussian mixture models (GMM) fail to account for the structure of variability of replicated data or repeated measures over time. Additionally, a priori cluster number assumptions add an additional complexity to the process. Current methods to optimize cluster labels and number can be inaccurate or computationally intensive for temporal gene expression data with this additional variability.

Results: An extension to a model-based clustering algorithm is proposed using mixtures of mixed effects polynomial regression models and the EM algorithm with an entropy penalized log-likelihood function (EPEM). The EPEM is used to cluster temporal gene expression data with this additional variability. The addition of random effects in our model decreased the misclassification error when compared to mixtures of fixed effects models or other methods such as K-Means and GMM. Applying our method to microarray data from a fracture healing study revealed distinct temporal patterns of gene expression.

Availability and Implementation: <https://github.com/darlenelu72/EPEM-GMM>.

Contact: dlu7@bu.edu

Supplementary information: [Supplementary data](#) are available at *Bioinformatics* online.

1 Introduction

Time is an intrinsic factor in almost all biological processes and of particular importance in the areas of development, pathology and aging. Leveraging temporal information of large datasets would be informative to all of these processes. Decreasing costs of microarray experiments have allowed investigators to conduct more elaborate designs (measurements taken over several time-points). However, microarray experiments are subject to a degree of variability in the measurements, therefore, biological replicates using replicated arrays are often needed to improve the stability of gene expression estimation. Ignoring the covariance structure between these repeated measurements over time or biological replicates could result in a failure to capture important sources of variability from the experiment leading to incorrect inferences or groupings of the genes.

This work is motivated by several prior microarray gene expression studies. These studies aimed to investigate the impacts of genetic and environmental variability on the post natal regenerative processes of fracture healing (Grimes *et al.*, 2011; Wigner *et al.*, 2010). Three strains of mice were randomly assigned to a regular diet versus a phosphate deficient diet. Phosphate deficiency leads to a rachitic state that delays the progression of bone healing. Three replicates of fracture calluses were obtained at each of the nine time-points from day 3 to 35, and each array contains measurements for about 20 000 genes. Variability exists between the repeated measurements over time and between the biological replicates at each time point. Most traditional clustering methods do not account for this variability.

Two commonly used clustering methods are K-Means (Hartigan and Wong, 1979) and hierarchical clustering (Sneath and

Sokal, 1973). However, these methods are implemented on multivariate data ignoring the temporal relationship between repeated measurements at each time-point of the same gene. In the family of model-based clustering, a probability model based approach assumes that data is generated from a mixture model of probability distributions. Examples include (i) a Gaussian mixture model (GMM) allowing each cluster to be represented by its own mean vector and covariance matrix (Fraley *et al.*, 2012), (ii) mixtures of fixed effects (FE) polynomial regression models (PRM) for temporal data [an extension to the GMM where each cluster is represented by parameters defined by a polynomial curve (Gaffney and Smyth, 1999)], and (iii) a further extension to mixtures of mixed effects (ME) linear regression models (Celeux *et al.*, 2005; Ng *et al.*, 2006) to account for correlation between biological replicates or between genes in temporal or cross-sectional data.

In these model-based approaches, gene-expression profiles in each cluster are assumed to be generated from one component of the mixture model. A popular method to estimate the mixture parameters is by maximum likelihood estimation (MLE) using the Expectation-Maximization (EM) algorithm (Dempster *et al.*, 1977). However, the EM algorithm is prone to optimization issues resulting from its sensitivity to initialization values (Biernacki *et al.*, 2003). In previous studies, the ME model was found to perform better than GMM when variability from biological replicates was present in the data (Celeux *et al.*, 2005). However, with an unknown cluster number, component number optimization is an additional step to the process. A common solution to this problem is to use model selection criteria such as Akaike's Information Criterion (AIC) (Aike, 1974), Bayesian Information Criterion (BIC) (Fraley *et al.*, 2012; Roeder *et al.*, 1997) or Integrated Complete Likelihood (ICL) (Biernacki *et al.*, 2000; McLachlan and Peel, 2000). While shown to perform well in estimating the number of components (Fonesca and G.M.S., 2007), these methods can be computationally intensive, especially in high-dimensional data with a large number of clusters or genes (Fraley *et al.*, 2012).

To counteract the computational inefficiency, a penalty term was proposed to be added to the objective function allowing the algorithm to learn at each iteration and discard illegitimate clusters (Yang *et al.*, 2012). We refer to this as the Entropy Penalized EM (EPEM) algorithm, which can be used to obtain the optimal component size and class labels of each gene in one-step. However, it was only formulated to be applied to multivariate data with no assumptions on the functional relationships in temporal data. Chamroukhi (2016) extended the EPEM to apply it to temporal data by fitting mixtures of fixed effects (FE) polynomial or spline regression models. The algorithm was compared to K-means and the GMM and found that his algorithm performed much better with a lower misclassification error rate (MCE). However, the performance was only reported for curves with a large number of observations (>50). Additionally, considerations for high-dimensional data or data with multiple replicates per gene were not assessed.

By using a ME mixture model (Celeux *et al.*, 2005) and entropy penalized maximum likelihood estimation (Yang *et al.*, 2012), we propose an extension to the clustering algorithm (Chamroukhi, 2016) for temporal data with additional sources of variability (from repeated measurements over time and biological replicates) and an unknown component number. Different underlying ME PRMs will be considered and compared to the models proposed in Yang *et al.* (2012) and Chamroukhi (2016). Using the best-performing ME PRM, the new EPEM algorithm will be compared to K-Means and GMM, which both use a two-step approach using BIC to optimize component number.

2 Materials and methods

Assume that the data consist of N temporal gene-expression profiles. Each profile consists of gene-expression measurements obtained at T time points, represented by y_i where $i = 1, \dots, N$. Assume that for each time-point we have R replicates so that each temporal gene-expression profile has $T \times R$ measurements. The goal is to cluster the N temporal gene-expression profiles into K groups using Gaussian finite mixture models.

2.1 Standard formulation of Gaussian mixture models

First, let us assume that $R = 1$ resulting in T observations for each gene (dropping the r index). GMMs assume that temporal gene-expression profiles from each cluster are generated from one component of a mixture of Multivariate Normal distributions. The k th cluster ($k = 1, \dots, K$) has a probability of occurrence, $\alpha_k = P(z_i = k)$, where z_i is the missing cluster label for the temporal profile of gene i and $\sum_{k=1}^K \alpha_k = 1$. If each cluster has its own set of parameters $\theta_k = \{\mu_k, \sigma_k^2\}$, a conditional probability density function (pdf) can be defined for y_i by the $N(\mu_k, \sigma_k^2)$ pdf (assuming expression values for gene i originated from cluster k). If the full set of parameters for the mixture model is $\Theta = \{\theta_1, \dots, \theta_K, \alpha_1, \dots, \alpha_K\}$, then the unconditional pdf for y_i can be represented as a weighted sum of the $N(\mu_k, \sigma_k^2)$ pdfs, $p_k(y_i|\theta_k)$, (1).

$$p(y_i|\Theta) = \sum_{k=1}^K \alpha_k p_k(y_i|\theta_k) \quad (1)$$

Therefore, the log-likelihood function based on the observed data $y = (y_1, \dots, y_N)'$ is given by

$$l(\Theta) = \log L(\Theta) = \sum_{i=1}^N \log \left\{ \sum_{k=1}^K \alpha_k p_k(y_i|\theta_k) \right\}. \quad (2)$$

Maximizing this log-likelihood function (2) is not trivial and a closed-form solution does not exist (McLachlan and Krishnan, 2008). However, if the problem can be formulated in a missing data framework, the EM algorithm can be used by assuming that the cluster memberships (z_i) are unknown (Dempster *et al.*, 1977). Let z_i consist of a vector of zero-one indicator variables, $(z_{i1}, \dots, z_{iK})$, where z_{ik} equals one if y_i originated from component k of the mixture model. Defining the complete data vector to be (y_i, z_i) , leads to the formulation of the complete-data log-likelihood as

$$l_c(\Theta) = \sum_{i=1}^N \sum_{k=1}^K z_{ik} \log \alpha_k + \sum_{i=1}^N \sum_{k=1}^K z_{ik} \log p_k(y_i|\theta_k). \quad (3)$$

The EM-algorithm (Dempster *et al.*, 1977) can then be used to determine the maximum likelihood estimates (MLE) for each of the parameters in Θ . The usefulness of this formulation is that, at each iteration, we are now able to obtain closed-form solutions of our parameters.

Extending this formulation to account for a functional relationship between y_i and the T -time points at which it was measured (x_i), can be done by using PRMs [i.e. $y_i = \beta_0 + \beta_1 x_i + \dots + \beta_p x_i^p + \epsilon_i$ (Gaffney and Smyth, 1999)]. Assuming normality, $(y_i|x_i)$ is now defined with a $N(x_i\beta_k, \Sigma_k)$ pdf, and the problem becomes clustering a mixture of PRMs that can be solved via the EM algorithm.

2.2 Mixed-effects model with gene- and replicate-specific random effects

Now assume that for each time-point we have R replicates, where each temporal gene-expression profile has $T \times R$ measurements. We

want to account for the variability between the biological replicates at each time point and allow the temporal trend for each gene to deviate from its cluster-specific temporal trend, yet still exhibit the same underlying pattern that distinguishes that cluster from the rest. A ME model does just that when the genes within a cluster are allowed to have their own intercept and/or slope (Celeux et al., 2005; Ng et al., 2006). Equation (4) includes a replicate-specific RE as well as time-varying gene-specific RE, where p and q are the FE and RE polynomial orders, respectively.

$$y_{trik} = \beta_{0k} + \beta_{1k}x_{trik} + \dots + \beta_{pk}x_{trik}^p + b_{0ik} + b_{1ik}x_{trik} + \dots + b_{qik}x_{trik}^q + c_{rik} + \epsilon_{trik} \quad (4)$$

$\beta_k = (\beta_{0k}, \beta_{1k}, \dots, \beta_{pk})$ are the k th cluster FE coefficients; x_{tri} is t th time-point for the r th replicate of the i th temporal gene-expression profile; $b_{ik} = (b_{0ik}, b_{1ik}, b_{qik})^T \sim N(0, G_k)$ is the gene-specific RE; $c_{rik} \sim N(0, \tau_k^2)$ is replicate-specific RE (between-replicate variability); $\epsilon_{trik} \sim N(0, \sigma_k^2)$ is the measurement error (within-replicate variability). Assume that b_{ik} , c_{rik} and ϵ_{trik} are mutually independent and each cluster is allowed to have a different mean vector and covariance matrix. The ME models considered, Eq0r1, Eq1r1 and Eq2r1, have parameters $\beta_{0k}, \beta_{1k}, \dots, \beta_{pk}, \sigma_k^2, G_k, \tau_k^2$, where G_k is a $(q+1) \times (q+1)$ matrix and $q = 0, 1$ and 2 , respectively. Lastly, the Eq2r0 models ignore the variability between replicates and include parameters $\beta_{0k}, \beta_{1k}, \dots, \beta_{pk}, \sigma_k^2, G_k$, where G_k is a 3×3 matrix. These models differ by the number of REs that are allowed.

The REs can be viewed as unobserved missing data and the EM algorithm (Dempster et al., 1977; McLachlan and Krishnan, 2008) can be used to determine the MLEs by iteratively maximizing the conditional expectation of the complete data log-likelihood (5),

$$\begin{aligned} l_c(\Theta) &= \log L(\Theta|y, b, c, z) \\ &= \log \prod_{i=1}^N \prod_{k=1}^K [\alpha_k p_k(y_i|b_i, c_i, \theta_k) p_k(b_i|\theta_k) p_k(c_i|\theta_k)]^{z_{ik}} \\ &= \sum_{i=1}^N \sum_{k=1}^K z_{ik} \log \alpha_k + \sum_{i=1}^N \sum_{k=1}^K z_{ik} \log [p_k(y_i|b_i, c_i, \theta_k)] \\ &\quad + \sum_{i=1}^N \sum_{k=1}^K z_{ik} \log [p_k(b_i|\theta_k)] + \sum_{i=1}^N \sum_{k=1}^K z_{ik} \log [p_k(c_i|\theta_k)] \end{aligned} \quad (5)$$

where $p_k(y_i|b_i, c_i, \theta_k)$, $p_k(b_i|\theta_k)$ and $p_k(c_i|\theta_k)$ are conditional Normal pdfs defined in the next section.

2.3 Entropy penalized maximum likelihood estimation

Several problems still exist with GMM clustering algorithm. Initialization of the EM algorithm is important, which can yield different results when initialized at different locations of the parameter space (Biernacki et al., 2003). To solve this, many algorithms require multiple random initializations and choosing the results for the one with the largest log-likelihood or initializations from other clustering methods such as K-Means (Figueiredo et al., 2002). However, this can be computationally intensive, particularly for large datasets. Additionally, the number of components in the standard EM needs to be specified. To choose the optimal number of components, many algorithms adapt a brute-force method by clustering the data over a range of cluster numbers and selecting the optimal clustering based on some criterion. This can be time consuming and limited to the range of cluster numbers considered.

To avoid this, a penalty term (Yang et al., 2012) is added in the same spirit as (Chamroukhi, 2016) to the standard log-likelihood function (5). The number of clusters in our data can be represented by the unknown class labels (z_i) for each temporal gene expression

profile. As the number of clusters in our data increase, making the model more complex, the entropy of z_i also increases. Using the entropy of z_i , $H(z_i)$, as the penalty term in our penalized log-likelihood maximization function allows for penalization of overly complex models. The derivation of $H(z_i)$ and $H(z)$ is shown below, where $z = (z_1, \dots, z_N)$.

$$\begin{aligned} H(z_i) &= E[-\log(P(z_i))] \\ &= -\sum_{k=1}^K P(z_i = k) \log[P(z_i = k)] = -\sum_{k=1}^K \alpha_k \log \alpha_k \\ H(z) &= E[-\log(P(z))] \\ &= -\sum_{i=1}^N \sum_{k=1}^K \alpha_k \log \alpha_k = -N \sum_{k=1}^K \alpha_k \log \alpha_k \end{aligned} \quad (6)$$

Adding the penalty term, $H(z)$ to (5), the new penalized log-likelihood (7) is obtained.

$$l_{c(p)}(\Theta) = l_c(\Theta) - \lambda H(z) \quad (7)$$

Maximum likelihood estimation (EM Algorithm)

Each probability function of (5) or (7) can be defined with normally distributed pdfs; $(y_i|b_i, c_i, \theta_k) \sim N(X\beta_k + Ub_{ik} + Vc_{ik}; \sigma_k^2 I_{TR})$; $(b_{ik}|\theta_k) \sim N(0, G_k)$; $(c_{ik}|\theta_k) \sim N(0, \tau_k^2 I_{TR})$, where X is the Vandermonde matrix of time-points of order p , U is a Vandermonde matrix of time-points of order q , and V is a diagonal matrix of size $TR \times R$ where the diagonals are unit vectors of length R . A simplified description of the EM Algorithm is presented here, see supplementary text for details.

E-step The E-step (Expectation step) of the EM algorithm takes the expectation of the complete data-log likelihood (7) conditional on the observed data and the parameter space of the previous iteration, resulting in the following objective function (McLachlan and Krishnan, 2008):

$$\begin{aligned} J(\Theta, \Theta^{(s)}) &= \sum_{i=1}^N \sum_{k=1}^K w_{ik}^{(s)} \log \alpha_k \\ &\quad + \sum_{i=1}^N \sum_{k=1}^K w_{ik}^{(s)} \left\{ -\frac{TR \log \sigma_k^2}{2} - \frac{\mathbb{E}[D_i^T D_i | y_i, \Theta^{(s)}]}{2\sigma_k^2} \right\} \\ &\quad + \sum_{i=1}^N \sum_{k=1}^K w_{ik}^{(s)} \left\{ -\frac{\log |G_k|}{2} - \frac{\mathbb{E}[b_{ik}^T G_k^{-1} b_{ik} | y_i, \Theta^{(s)}]}{2} \right\} \\ &\quad + \sum_{i=1}^N \sum_{k=1}^K w_{ik}^{(s)} \left\{ -\frac{R \log \tau_k^2}{2} - \frac{\mathbb{E}[c_{ik}^T c_{ik} | y_i, \Theta^{(s)}]}{2\tau_k^2} \right\} \\ &\quad + \lambda \sum_{i=1}^N \sum_{k=1}^K \alpha_k \log \alpha_k \end{aligned} \quad (8)$$

where $D_i = y_i - X\beta_k - Ub_{ik} - Vc_{ik}$ and $w_{ik}^{(s)}$ is the posterior probability that the temporal gene-expression profile belongs to the k th cluster conditioning on the observed data and the current parameter estimates $\Theta^{(s)}$. The updating equation for $w_{ik}^{(s)}$ is given in (9) (Chamroukhi, 2016; McLachlan and Peel, 2000)

$$\begin{aligned} \hat{w}_{ik}^{(s)} &= \mathbb{E}[z_{ik} | y_i, \Theta^{(s)}] \\ &= \frac{P(y_i | z_i = k, \Theta^{(s)}) P(z_i = k)}{P(y_i | \Theta^{(s)})} \\ &= \frac{\alpha_k^{(s)} p_k(y_i | \theta_k^{(s)})}{\sum_{k=1}^K \alpha_k^{(s)} p_k(y_i | \theta_k^{(s)})} \end{aligned} \quad (9)$$

To complete the E-step, conditional expectations $(E[b_{ik}|y_i], E[c_{ik}|y_i], E[b_{ik}b_{ik}^T|y_i]$ and $E[c_{ik}c_{ik}^T|y_i])$ need to be determined.

M-step The M-step involves maximizing (8) with respect to each of the parameters $(\alpha_k, \beta_k, \sigma_k^2, \tau_k^2, G_k)$ by taking the derivative with respect to each parameter. The formulas for the MLEs of each parameter is given in (10) (to solve for α_k , we must use a Lagrange multiplier to satisfy the constraint that $\sum_{k=1}^K \alpha_k = 1$).

$$\begin{aligned}\hat{\beta}_k^{(s+1)} &= \frac{\sum_{i=1}^N w_{ik}^{(s)} [\hat{\beta}_k^{(s)} + \hat{\sigma}_k^{2(s)} (X'X)^{-1} X' \Gamma_k^{-1(s)} (y_i - X \hat{\beta}_k^{(s)})]}{W_i} \\ \hat{\tau}_k^{2(s+1)} &= \frac{1}{R W_i} \sum_{i=1}^N w_{ik}^{(s)} \left[\hat{\tau}_k^{4(s)} (y_i - X \hat{\beta}_k^{(s)})' \Gamma_k^{-1(s)} V V' \Gamma_k^{-1(s)} \right. \\ &\quad \left. (y_i - X \hat{\beta}_k^{(s)}) + R \hat{\tau}_k^{2(s)} - \hat{\tau}_k^{4(s)} \text{tr}(\Gamma_k^{-1(s)} V V') \right] \\ \hat{G}_k^{(s+1)} &= \frac{1}{W_i} \sum_{i=1}^N w_{ik}^{(s)} \left[\hat{G}_k^{(s)} U' \Gamma_k^{-1(s)} (y_i - X \hat{\beta}_k^{(s)}) (y_i - X \hat{\beta}_k^{(s)})' \right. \\ &\quad \left. \Gamma_k^{-1(s)} U \hat{G}_k^{(s)} + \hat{G}_k^{(s)} - \hat{G}_k^{(s)} U' \Gamma_k^{-1(s)} U \hat{G}_k^{(s)} \right] \\ \hat{\sigma}_k^{2(s+1)} &= \frac{1}{T R W_i} \sum_{i=1}^N w_{ik}^{(s)} \left[\hat{\sigma}_k^{4(s)} (y_i - X \hat{\beta}_k^{(s)})' \Gamma_k^{-1(s)} \Gamma_k^{-1(s)} \right. \\ &\quad \left. (y_i - X \hat{\beta}_k^{(s)}) + \text{tr}(\text{Cov}(D_{ik} | y_i, \Theta^{(s)})) \right] \\ \hat{\alpha}_k^{(s+1)} &= \frac{W_i}{N} + \lambda^{(s)} \hat{\alpha}_k^{(s)} \left\{ \log \hat{\alpha}_k^{(s)} - \sum_{k=1}^K \hat{\alpha}_k^{(s)} \log \hat{\alpha}_k^{(s)} \right\}\end{aligned} \quad (10)$$

where $W_i = \sum_{k=1}^K w_{ik}^{(s)}$, $\Gamma_k^{-1} = (U G_k U' + \tau_k^2 V V' + \sigma_k^2 I_{TR})^{-1}$ and $\text{Cov}(D_{ik} | y_i, \Theta^{(s)}) = U \left[\hat{G}_k^{(s)} - \hat{G}_k^{(s)} U' \Gamma_k^{-1(s)} U \hat{G}_k^{(s)} \right] U' + V \left[\hat{\tau}_k^{2(s)} I_R - \hat{\tau}_k^{4(s)} V' \Gamma_k^{-1(s)} V \right] V'$.

λ is also estimated in an iterative way. $\lambda \in [0, 1]$ controls how much of a role the penalty term plays in the objective function. If λ is close to zero the penalized log-likelihood function is close to the standard log-likelihood function (5) and more stable, exhibiting convergence properties of the standard EM. If λ is large (i.e. close to 1), the penalty term plays a larger role by creating competition between clusters. It forces the algorithm to get rid of illegitimate clusters with small proportions. For cluster k , if $\log \hat{\alpha}_k^{(s)} - \sum_{k=1}^K \hat{\alpha}_k^{(s)} \log \hat{\alpha}_k^{(s)} > 0$, then $H(z)$ is small and the second term is positive, causing $\hat{\alpha}_k^{(s+1)}$ to increase. However, if $\log \hat{\alpha}_k^{(s)} - \sum_{k=1}^K \hat{\alpha}_k^{(s)} \log \hat{\alpha}_k^{(s)} < 0$, then $\hat{\alpha}_k^{(s+1)}$ will decrease. If it is lower than a set threshold, the cluster will be deemed illegitimate and removed by the algorithm. Note that this penalty term does not guarantee an increased likelihood function at each iteration of the algorithm. However, through simulation studies it was shown that the penalized log-likelihood function levels off after convergence of the algorithm (Chamroukhi, 2016). $\lambda^{(s+1)}$ is determined by $\hat{\alpha}_k^{(s)}$, $\hat{\alpha}_k^{(s+1)}$ and $\hat{w}_{ik}^{(s)}$, defined in (11).

$$\lambda^{(s+1)} = \min \left\{ \frac{\sum_{k=1}^K \exp(-\eta | \log \hat{\alpha}_k^{(s+1)} - \log \hat{\alpha}_k^{(s)} |)}{K}, \frac{\left(1 - \max_k \frac{\sum_{i=1}^N w_{ik}^{(s)}}{n} \right)}{-\max_k \hat{\alpha}_k^{(s-1)} \sum_{k=1}^K \hat{\alpha}_k^{(s)} \log \hat{\alpha}_k^{(s)}} \right\} \quad (11)$$

where η is set to be $\min \left\{ 1, 0.5^{\lfloor \frac{T}{2} - 1 \rfloor} \right\}$. The motivation and derivation of λ is presented in much detail in Yang *et al.* (2012) and Chamroukhi (2016). λ is updated so that it is large when $\hat{\alpha}_k$ is not changing enough between iterations to promote cluster competition. However, when $\hat{\alpha}_k$ exhibits a large change, λ needs to be small to promote stability of the algorithm.

2.4 Algorithm

2.4.1 Initialization

To initialize the algorithm, the number of clusters in the mixture model is set to equal the number of temporal gene-expression profiles in the dataset. To obtain cluster-specific parameter estimates, a

FE PRM is fit to the temporal-gene expression profiles for each gene to obtain $\beta_k^{(0)}$ for each cluster k . For all clusters, $\sigma_k^{2(0)}$ and $\tau_k^{2(0)}$ are initialized to be the median residual error from all gene- or replicate-specific models run (to avoid singularities when the covariance matrix is too small), respectively. $G_k^{(0)}$ is initiated as $\sigma_k^{2(0)} \times (X'X)^{-1}$ and $\alpha_k^{(0)} = 1/N$. The posterior probability ($w_{ik}^{(0)}$) is initialized by $\lambda = 0$, $\alpha_k^{(0)}$, $\beta_k^{(0)}$, $\sigma_k^{2(0)}$, $G_k^{(0)}$ and $\tau_k^{2(0)}$. Lastly, the order of the fixed (p) and mixed (q) effects polynomial order must be specified prior to running the algorithm. The same polynomial order is fit to all clusters. Model selection methods such as cross-validation may be used to select the optimal order (Gaffney, 2004).

2.4.2 Iteration s of the EPEM algorithm

For iteration s , the maximization step updates the parameters in our model defined in (10). At each subsequent iteration, the algorithm alternates between the E- and M-step to remove illegitimate clusters with small $\alpha_k < 1/N$. After stability of cluster number is reached (no clusters are removed for 60 consecutive iterations), λ is set to zero and convergence of the algorithm can proceed and is assessed by the FE coefficients such that $\max_k \|\beta_k^{(s)} - \beta_k^{(s+1)}\| < \epsilon = 10^{-4}$. At the last iteration, predicted cluster labels ($\hat{z}_i$) is determined by setting membership to be the cluster with the largest $\hat{w}_{ik}$ over all k for each gene ($\hat{z}_i = \text{argmax}_k \hat{w}_{ik}$). The algorithm is described in detail in Table 1.

2.5 Simulations

Simulations were performed to assess the performance of each ME considered by measuring the misclassification error (MCE) of each clustering. MCE is defined as the error rate for a given cluster label relative to the known truth. Data was generated with four clusters with uniform ($\alpha_k = 0.25$) and varying ($\alpha_k \in (0.4, 0.3, 0.2, 0.1)$) mixing proportions, varying sample sizes ($N = 200, 500$ or 5000), varying replicates (2, 4, 10), and nine different within- and between-replicate variability (σ_k^2, τ_k^2) specifications with varying orders (all

Table 1. EM-Algorithm for entropy-penalized maximum likelihood estimation

1. **Initialize EM algorithm:** Start with $K^{(0)} = N$ different clusters
 - a. Compute $w_{ik}^{(0)}$ from $\beta_k^{(0)}$, $\tau_k^{(0)}$, $G_k^{(0)}$, $\sigma_k^{2(0)}$ and $\alpha_k^{(0)} = 1/N$
 - b. Let $\lambda^{(0)} = 1$ and converge = 0
2. **For $s = 1$:**
 - a. Compute $\alpha_k^{(1)}$ from $w_{ik}^{(0)}$, $\lambda^{(0)}$ and $\alpha_k^{(0)}$
 - b. Compute $\lambda^{(1)}$ from $\alpha_k^{(1)}$, $\alpha_k^{(0)}$ and $w_{ik}^{(0)}$
 - c. Update $K^{(1)} = K^{(0)}$ - number of components where $\alpha_k^{(1)} \leq 1/N$
 - d. Normalize $w_{ik}^{(0)}$ and $\alpha_k^{(1)}$ to satisfy the sum to 1 constraint
 - e. Compute $\beta_k^{(1)}$, $\tau_k^{2(1)}$, $G_k^{(1)}$, $\sigma_k^{2(1)}$ from $w_{ik}^{(0)}$
 - f. Compute $w_{ik}^{(1)}$ from $\beta_k^{(1)}$, $\tau_k^{2(1)}$, $G_k^{(1)}$, $\sigma_k^{2(1)}$ and $\alpha_k^{(1)}$
3. **For $s \geq 2$:**
 - a. Compute $\alpha_k^{(s)}$ from $w_{ik}^{(s-1)}$, $\lambda^{(s-1)}$ and $\alpha_k^{(s-1)}$
 - b. Compute $\lambda^{(s)}$ from $\alpha_k^{(s)}$, $\alpha_k^{(s-1)}$ and $w_{ik}^{(s-1)}$
 - c. Update $K^{(s)} = K^{(s-1)}$ - # of components where $\alpha_k^{(s)} \leq 1/N$
 - d. If $s \geq 60$ and $K^{(s-60)} - K^{(s)} = 0$, then let $\lambda^{(s)} = 0$
 - e. Normalize $w_{ik}^{(s-1)}$ and $\alpha_k^{(s)}$ to satisfy the sum to 1 constraint
 - f. Compute $\beta_k^{(s)}$, $\tau_k^{2(s)}$, $G_k^{(s)}$, $\sigma_k^{2(s)}$ from $w_{ik}^{(s-1)}$
 - h. Compute $w_{ik}^{(s)}$ from $\beta_k^{(s)}$, $\tau_k^{2(s)}$, $G_k^{(s)}$, $\sigma_k^{2(s)}$ and $\alpha_k^{(s)}$
 - i. If $\max_k \|\beta_k^{(s)} - \beta_k^{(s-1)}\| < \epsilon$ then converge = 1 and STOP, else $s = s + 1$ and repeat step 3

Fig. 1. One iteration of a dataset generated for simulation studies A and B (4 replicates, 50 genes per cluster and 4 true clusters). ^aLow error: $\sigma_k^2=0.01$, $\tau_k^2=0.01$; High error: $\sigma_k^2=0.1$, $\tau_k^2=0.1$; ^b $\sigma_k^2 \in (0.01 - 0.03)$; $\tau_k = 0.01$

pairwise combinations of $\{0.01, 0.1, 0.4\}$ for varying degree of polynomial (Simulation A and Simulation B). Table 2 summarizes the parameter estimates used to generate the data for two types of simulated data. Simulation A data consists of four distinct clusters of linear and quadratic temporal gene-expression profiles with 10 time points, low ($\sigma_k^2 = 0.01$ and $\tau_k^2 = 0.01$) and high ($\sigma_k^2 = 0.1$ and $\tau_k^2 = 0.1$) error (panel 1 and 2 of Fig. 1, respectively). Simulation B data consists of more realistic cubic time-curves and 12 time points (panel 3 of Fig. 1).

To investigate the impact of high-dimensional data with many clusters, datasets with 30 clusters were generated with linear and quadratic curves (Supplementary Table S1; $N=6000$ with 200 genes per cluster; $\alpha_k = 1/30$; 4 replicates; 10 time points with 1 day intervals). The clustering algorithm (EPEM) was run for $p=2$ and $q=2$. Robustness of departures from normality was assessed with Simulation A parameters ($N=500$ with 125 genes per cluster; $\alpha_k = 0.25$; 4 replicates) where the between- and within-replicate variability is drawn from the Student's T-distribution (10 degrees of freedom; to allow for outliers) and the Gamma distribution ($b_{qik} \sim \text{Gamma}(3, 10^2)$, $c_{rik} \sim \text{Gamma}(3, 5)$, $e_{trik} \sim \text{Gamma}(3, 10)$; to allow for skewness). Supplementary Figure S1 presents a histogram, for cluster one by underlying distribution (Normal, Student's T and Gamma), of the expression values at day one from a simulated dataset.

1000 datasets were generated for each scenario. To assess the performance of each method, the predicted cluster number ($\hat{K}$) and MCE are reported. The ME models, Eq0r1, Eq1r1, Eq2r1 and Eq2r0, were compared between each other and additionally with the FE only model proposed in Chamroukhi (2016) (Eq0r0) and the multivariate clustering in Yang et al. (2012) (NOM). For NOM, Eq0r0 and Eq2r0 models, the clustering algorithm was run on the mean of the measurements over all replicates of each time point (each temporal gene-expression profile has 10 expression measurements). Data were generated from curves defined by (4), where $(p, q) \in \{(1, 1), (2, 2), (3, 3)\}$, corresponding to linear, quadratic and cubic polynomial functions, respectively. The clustering algorithm for simulation A and B was run with $p=2$ or 3, respectively, and varying q (0 to 2). $q \leq 2$ was chosen for pragmatic purposes in an attempt to not over-specify the model as model based clustering requires the estimation of a large number of parameters.

The EPEM algorithm with ME models were compared with two easily implemented methods in R, K-Means clustering and GMM using the Stats and MClust package in R (Fraley et al., 2012), respectively. A two-step approach was necessary for KMeans and GMM, where multiple clusterings must be performed

Table 2. Simulation A and B parameters to obtain datasets used in simulation studies with uniform ($\alpha_k=0.25$) or varying ($\alpha_k \in (0.4, 0.3, 0.2, 0.1)$) mixing proportions for each cluster

	Parameter	1	2	3	4
Sim A ^a	β_0	0.5	0.1	0.5	-1
	β_1	-	0.5	-0.1	1.5
	β_2	-	-	-	-0.1
	β_3	-	-	-	-
	$\text{Var}(b_0)$	$0.01\sigma^2$	$0.02\sigma^2$	$0.1\sigma^2$	$0.2\sigma^2$
	$\text{Var}(b_1)$	-	$0.1\sigma^2$	$0.02\sigma^2$	$0.3\sigma^2$
	$\text{Var}(b_2)$	-	-	-	$0.02\sigma^2$
	$\text{Var}(b_3)$	-	-	-	-
	σ^2	0.01, 0.10 or 0.40			
	τ^2	0.01, 0.10 or 0.40			
Sim B ^b	β_0	-1.05	-2.3	2.03	2.85
	β_1	0.41	0.5	-0.58	-0.5
	β_2	-0.026	-0.02	0.033	0.024
	β_3	$4.00\text{E}-4$	$2.50\text{E}-4$	$-5.03\text{E}-4$	$-3.56\text{E}-4$
	$\text{Var}(b_0)$	0.21	0.46	0.406	0.57
	$\text{Var}(b_1)$	0.02	0.03	0.03	0.03
	$\text{Var}(b_2)$	$5.20\text{E}-4$	$4.00\text{E}-4$	$6.60\text{E}-4$	$4.80\text{E}-4$
	$\text{Var}(b_3)$	$8.00\text{E}-6$	$5.00\text{E}-6$	$1.00\text{E}-5$	$7.20\text{E}-6$
	σ^2	0.01	0.02	0.01	0.03
	τ^2	0.01			

Note: Datasets were generated with varying sample size (200, 500, 5000), replicates (2, 4 or 10) and within- (σ_k^2) or between- (τ_k^2) replicate variability.

Simulation A had 10 time points with 1 day intervals (1, 2, ..., 10 days). Simulation B had 10 time points with 3 day intervals (1, 4, 7, ..., 34 days).

^aCorrelation (ρ) between b_i such that $\rho(b_0, b_1) = -0.5$, $\rho(b_0, b_2) = 0.4$ and $\rho(b_1, b_2) = -0.9$.

^b $\rho(b_0, b_1) = -0.7$, $\rho(b_0, b_2) = 0.6$, $\rho(b_1, b_2) = -0.8$, $\rho(b_0, b_3) = 0.4$, $\rho(b_1, b_3) = -0.6$, $\rho(b_2, b_3) = 0.9$.

over a range of component sizes (2–60) and using BIC to select the best set of class labels. $\hat{K}$, MCE and minutes of run time (Linux Centos 6.6 operating system) were compared between the 3 algorithms.

2.6 Application

Animal research was conducted in conformity with an IACUC approved protocol. Aj strain of mice were purchased from Jackson Laboratories (Bar Harbor, ME). All fracture studies were performed on 8 to 10 week old male mice as previously described (Jepsen et al., 2008), days 3, 5, 7, 10, 14, 18, 21, 28 and 35. Total RNA was extracted and prepared for microarray analysis as previously described (Bais et al., 2009). All procedures were performed at Boston University Microarray Resource Facility exactly as described in GeneChip® Whole Transcript (WT) Sense Target Labeling Assay Manual (Affymetrix, Santa Clara, CA, current version available at www.affymetrix.com). See supplementary text for additional details. For illustration purposes, we applied the EPEM-Eq0r0 ($p=3$; no REs) or Eq2r0 ($p=3$; $q=2$; no replicate-specific random effect) clustering algorithm to the bone healing study with data from strain AJ mice on the control diet. The expression values were standardized by the gene-specific mean and standard deviation. First, a differential expression analysis was used to subset the data to include differentially expressed genes. The EDGE package (Storey et al., 2005) was used to fit a null model versus a full model (third order polynomial). A likelihood ratio test was performed to determine if the full model was a better fit than the null model using an alpha-level of 0.01. All simulations and results were obtained using R software (<http://www.r-project.org>).

Table 3. Simulation Results. EPDM clustering results with different underlying models (data with 4 clusters, 4 replicates and 50, 125 or 1250 genes per cluster) compared to KMeans and GMM clustering

Data	Error ^a	Model	50 Genes/Cluster		125 Genes/Cluster		1250 Genes/Cluster	
			$\hat{K}$ (SD)	MCE (SD)%	$\hat{K}$ (SD)	MCE (SD)%	$\hat{K}$ (SD)	MCE (SD)%
Simulation A (Linear or Quadratic Curves)	Low	EPDM-NoM (No FE or RE)	6.3 (0.8)	24.5 (8.7)	8.3 (0.9)	41.3 (6.4)	15.4 (1.5)	59.5 (2.4)
		EPDM-Eq0r0 (No RE)	6.7 (1.1)	28.1 (10.5)	8.6 (1.0)	43.3 (6.2)	15.3 (1.5)	59.2 (2.9)
		EPDM-Eq0r1 ($p=2, q=0$)	5.2 (0.7)	12.6 (6.7)	6.0 (0.5)	19.5 (4.3)	6.9 (0.8)	22.9 (1.7)
		EPDM-Eq1r1 ($p=2, q=1$)	4.3 (0.5)	2.7 (4.7)	4.2 (0.4)	2.1 (4.2)	4.4 (0.5)	4.7 (5.3)
		EPDM-Eq2r1 ($p=2, q=2$)	4.2 (0.5)	2.2 (4.4)	4.1 (0.3)	0.8 (2.7)	4.2 (0.4)	2.2 (4.2)
		EPDM-Eq2r0 ($p=2, q=2$, No Rep RE ^b)	4.3 (0.5)	2.7 (5.3)	4.0 (0.2)	0.2 (1.7)	4.1 (0.3)	0.7 (2.7)
		KMeans	3.0 (0.0)	25.0 (0.0)	10.0 (0.0)	31.0 (1.6)	11.0 (0.0)	32.5 (0.7)
		GMM	4.2 (0.4)	1.7 (3.6)	4.8 (0.7)	7.4 (6.1)	7.8 (0.4)	20.8 (1.2)
		EPDM-NoM (No FE or RE)	6.8 (1.1)	30.3 (10.4)	8.7 (1.0)	44.5 (5.9)	15.4 (1.5)	59.8 (2.4)
		EPDM-Eq0r0 (No RE)	6.7 (1.1)	29.5 (10.3)	8.6 (1.0)	44.0 (5.9)	15.4 (1.6)	59.7 (2.8)
Simulation A (Linear or Quadratic Curves)	High	EPDM-Eq0r1 ($p=2, q=0$)	5.2 (0.7)	13.0 (7.0)	6.0 (0.5)	19.8 (4.2)	7.0 (0.8)	23.0 (1.7)
		EPDM-Eq1r1 ($p=2, q=1$)	4.3 (0.5)	2.7 (4.7)	4.2 (0.4)	2.0 (4.1)	4.5 (0.5)	5.0 (5.5)
		EPDM-Eq2r1 ($p=2, q=2$)	4.3 (0.5)	2.6 (4.5)	4.1 (0.3)	1.0 (3.0)	4.2 (0.4)	1.9 (3.9)
		EPDM-Eq2r0 ($p=2, q=2$, No Rep RE ^b)	4.3 (0.5)	3.2 (6.0)	4.0 (0.2)	0.6 (3.4)	4.1 (0.3)	0.9 (3.0)
		KMeans	3.0 (0.2)	26.1 (0.8)	9.0 (0.1)	36.8 (4.2)	16.8 (5.3)	58.8 (12.6)
		GMM	3.5 (0.7)	15.6 (11.4)	4.8 (0.6)	8.4 (6.0)	8.1 (1.6)	22.9 (2.1)
Simulation B (Cubic Curves)	Mixed	EPDM-NoM (No FE or RE)	6.3 (1.4)	26.6 (12.1)	8.2 (1.1)	43.3 (7.3)	33.6 (2.9)	84.3 (1.4)
		EPDM-Eq0r0 (No RE)	5.9 (1.1)	23.3 (11.2)	7.9 (0.8)	41.3 (6.9)	33.5 (3.2)	84.4 (1.4)
		EPDM-Eq0r1 ($p=3, q=0$)	6.6 (1.3)	28.8 (12.7)	8.3 (0.8)	44.4 (5.1)	20.5 (1.8)	69.6 (2.3)
		EPDM-Eq1r1 ($p=3, q=1$)	4.8 (0.7)	9.9 (8.1)	5.9 (0.8)	19.8 (7.9)	9.0 (1.2)	40.1 (7.4)
		EPDM-Eq2r1 ($p=3, q=2$)	4.4 (0.6)	4.7 (6.4)	4.6 (0.8)	4.5 (6.0)	4.4 (0.7)	2.3 (4.7)
		EPDM-Eq2r0 ($p=3, q=2$, No Rep RE ^b)	4.3 (0.5)	3.0 (5.4)	4.2 (0.4)	1.2 (3.1)	4.0 (0.1)	0.2 (1.3)
		KMeans	6.2 (2.1)	24.8 (4.9)	8.7 (0.8)	36.4 (8.2)	24.2 (0.8)	73.4 (1.7)
		GMM	4.0 (0.1)	0.0 (0.4)	4.0 (0.0)	0.0 (0.01)	5.4 (0.6)	14.6 (6.0)

Note: Average (SD) predicted cluster number ($\hat{K}$) and misclassification error (MCE %) over 1000 simulated datasets.

EPDM, Entropy Penalized EM Algorithm; FE, Fixed Effect; RE, Random Effect; GMM, Standard Gaussian Mixture Model; p, FE polynomial order; q, RE polynomial order.

^aLow: $\sigma_k^2=0.01, \tau_k^2=0.01$; High: $\sigma_k^2=0.1, \tau_k^2=0.1$; Mixed: $\sigma_k^2 \in (0.01 - 0.03); \tau_k = 0.01$.

^bNo Rep RE drops c_{rik} from (4).

3 Results

3.1 Varying the polynomial regression model of EPDM

The clustering results from Simulation A (linear and quadratic curves) and B (cubic curves) with uniform mixing proportions are shown in Table 3 for a fixed replicate number ($R=4$), varied number of genes per cluster (50, 125 and 1250), and different error assumptions (Simulation A, low: $\sigma_k^2=0.01, \tau_k^2=0.01$; high: $\sigma_k^2=0.1, \tau_k^2=0.1$; Simulation B, mixed: $\sigma_k^2 \in (0.01, 0.02, 0.01, 0.03)$, $\tau_k^2=0.01$).

In Simulation A, among the set of EPDM models, the ME models with $p=2$ and $q=2$, Eq2r1 and Eq2r0 (no replicate specific effect), produced the clusters with the lowest MCE ($<3.2\%$) in all scenarios considered. Not allowing for the additional heterogeneity within clusters (NoM or Eq0r0) results in a partition with too many clusters and high MCE. Averaging over replicates and dropping the replicate-specific effect (c_{rik}) did not substantially impact clustering results with similar accuracy in prediction of cluster labels between Eq2r1 and Eq2r0. In fact, in a majority of scenarios, averaging over replicates resulted in a slightly lower MCE. Results were similar for data with 50, 125 or 1250 genes per cluster. An increase in between- and within-replicate variability (high error: $\sigma_k^2=0.1$ and $\tau_k^2=0.1$) resulted in clusters with slightly higher MCE. Supplementary Figure S2, summarizes the results of Eq2r1 and Eq2r0 in simulation A for all nine underlying assumptions we considered (data with 125 genes per cluster and 4 replicates; results for other EPDM models showed similar trends to Table 3). For low ($\sigma_k^2=0.01$) and high ($\sigma_k^2=0.1$) within-replicate variability, an increase in between-replicate error

from 0.01 to 0.4 (τ_k^2 ; rows 1 and 2) did not impact the MCE. However, when the within-replicate variability was very high (0.4), an increase in the between-replicate variability resulted in an increase in MCE of the predicted class labels. Results were similar when we considered clustering higher order curves (Simulation B), however, clustering data averaged over the replicates (Eq2r0) decreased the MCE by more than 1.7% and performed with the lowest MCE regardless of the number of genes in each.

The estimated parameters for each cluster resulting from Eq0r0 and Eq2r1 models are given in Supplementary Tables S3 and S4 for data with 125 genes/cluster, $R=4$ and 4 true clusters. In both models, the algorithm did fairly well to estimate the fixed effects parameters (β_k) with low bias. Notice that an increase in predicted cluster number ($\hat{K}$) is proportional to a decrease in the estimated within-replicate variability (σ_k^2). Eq0r0 model underestimated the within-replicate variability (σ_k^2), resulting in more clusters. The Eq2r1 model was able to accurately estimate α_k, σ_k^2 and τ_k^2 . However, for simulation A, the covariance matrix for the gene-specific random effects appear to be biased, potentially attributed to over-specification of q for the linear-curves. In contrast, when the curves were purely cubic, the estimated random-effects are predicted quite well despite the under-specification of q .

3.2 Comparison to K-Means and GMM

Table 3 also compares KMeans, GMM and EPDM. KMeans performed poorly with MCE $> 26\%$ over all scenarios. GMM performed well for smaller datasets (50 or 125 genes/cluster), with an

almost perfect clustering in simulation B. However, for higher-dimension data with 1250 genes/cluster, we see that the MCE increases to more than 14%. Regarding computation time, KMeans had the fastest run-time (<3 min) over all scenarios considered. When clustering cubic curves, GMM exhibited the largest increase in run-time. Clustering a dataset with 5000 (1250 genes per cluster; 159 min) versus 400 (50 genes per cluster; 1 min) temporal gene expression profiles was 159 times longer. In contrast, the EPME-Eq2r1 was only 65 times longer (353 min versus 5 min). As N increases, there is evidence that the computation time does not increase as fast for EPME compared to GMM, most likely attributed to the two-step process when using GMM with an unknown cluster number, which we described previously. Additionally, GMM performed with a slightly elevated error compared to the mixed effects models (Eq2r1 or Eq2r0) when the within-replicate variability was in (0.01, 0.10) (Supplementary Fig. S2). However, when the within-replicate variability was very high, the MCE quickly increased to more than 20% suggesting that GMM does not work well when the variability is this high. Lastly, for high-dimensional data (30 clusters), the EPME-Eq2r1 resulted in clusters with a lower MCE $\pm$ SD compared to GMM ($11 \pm 2.6\%$ versus $22 \pm 4.4\%$, when averaged over 100 simulations).

3.3 Unequal mixing proportions

When the mixing proportions are no longer uniform [$\alpha_k \in (0.4, 0.3, 0.2, 0.1)$] as opposed to uniform (Table 3; Supplementary Fig. S2), similar results were observed for EPME models Eq2r1 and Eq2r0 (Supplementary Table S4). When the data consisted of cluster sizes of 80, 60, 40 and 20 genes per cluster, 3% of the iterations resulted with the covariance matrix of a cluster with very few genes (20 or 40 genes per cluster) to be close to singular, which was also consistent with (Yang et al., 2012). In contrast, for the previous scenario, GMM performs with lower MCE compared to either EPME models. However, when the cluster sizes increased to 200, 150, 100 or 50 genes per cluster, the two EPME models perform with higher accuracy than GMM.

3.4 Departures from normality

When the variabilities are sampled from the Student's T-distribution (Supplementary Table S5), only Eq2r0 and GMM perform with high precision (MCE = 0.7%). The other EPME models we considered are not shown in the table, but performed with similar trends as in Table 3. When the errors are simulated from a Gamma distribution, all models perform poorly with MCE >25%.

3.5 Effect of replicate number

When the effect of replicate number was assessed with the Eq2r1 model in simulation A, we found that for low-variability data ($\sigma_k^2=0.01$, $\tau_k^2=0.01$) an increase in replicate number did not decrease the MCE. In fact, a slight increase was observed. However, for high-variability data ($\sigma_k^2=0.1$, $\tau_k^2=0.1$), increasing the replicate number from 2 to 10 lowered the MCE from 8.2% to 2.7% or 1.3% to 0.8% in data with 50 or 125 genes per cluster, respectively.

3.6 Application to bone healing study

Of the 21 187 genes, 8934 genes were found to be differentially expressed. From the simulation results, we found that for cubic curves, Ep2r0 performed with the lowest MCE over all ME models considered. Therefore, we perform the EPME clustering algorithm with an underlying model specified Eq2r0 (12) and by Eq0r0 (no REs) for comparison.

$$y_{ik} = \beta_{0k} + \beta_{1k}x_i + \beta_{2k}x_i^2 + \beta_{3k}x_{iri}^3 + b_{0ik} + b_{1ik}x_i + b_{2ik}x_i^2 + \epsilon_{ik} \quad (12)$$

From Figure 2, 18 clusters were predicted from the algorithm with only FE (Eq0r0) compared to the 9 clusters predicted from Eq2r0 (Fig. 3). Several clusters obtained from Eq0r0 appear to be very similar (clusters 4, 5 and 18 or 13 and 17). When we include 2nd-order polynomial MEs, we obtain 9 clusters from Eq2r0 that are more distinct. Cluster 1, 2, 3 and 9 grouped genes that decreased their expression after bone fracture, where cluster 1, 2 and 9 appear to return back to a steady expression level after 28 days. Cluster 4, 5, 7 and 8 included genes that had an initial increase in gene expression after bone fracture followed by different rates of decline or location of change-point. Cluster 8 showed a relatively flat temporal pattern of gene expression. The smallest and largest cluster consisted of 179 temporal gene-expression profiles ($\alpha_k=0.02$) and 2501 genes ($\alpha_k=0.28$). The cluster-specific within-replicate variability (σ_k^2) varied from 0.06 to 0.36. Parameter estimates for Eq0r0 and Eq2r0 are found in Supplementary Tables S4 and S5, respectively. The adjusted rand index (ARI) (Hubert and Arabie, 1985) between the two sets of class labels is 0.6 suggesting weak agreement between the two results (an ARI of 1 corresponds to identical classifications).

4 Discussion

In this paper, we proposed an entropy penalized EM algorithm with mixtures of ME regression models to cluster data with additional variability (from biological replicates and repeated measurements over time). We found that the addition of the REs in our mixture model decreased the misclassification error by the clustering algorithm compared to mixtures of FE models (Chamroukhi, 2016) and other more popular methods (Fraleigh et al., 2012; Hartigan and Wong, 1979; Yang et al., 2012).

In data with high within- and between-replicate error, we found that having more replicates slightly lowered the MCE. However, the impact was minimal when the number of genes per cluster increased from 50 to 125. Furthermore, in all scenarios, no big differences in the accuracy of the predicted class labels were observed when we ignored the replicate-specific RE, c_{rik} (Eq2r1 versus Eq2r0), suggesting that averaging over replicates is an effective way to reduce the dimension of the data and prevent over-specifying the model without a sacrifice in accuracy.

When we assessed data with different cluster sizes, we found that clusters with very few number of genes could result in singularity issues in the covariance matrix estimation of the EPME algorithm, which is consistent to findings from Yang et al. (2012). Yang's solution was to use a constrained covariance matrix to circumvent this, which is an option we can implement in the future. However for the purposes of this manuscript with a primary application to gene expression clustering with thousands of genes, we believed that having large enough cluster sizes was a reasonable assumption to make.

In general, GMM performed well, however situations with a very high variability (i.e. $\sigma^2=0.4$), sample size (i.e. 1250 genes per cluster) or cluster number (i.e. $K=30$) resulted in classifications with reduced accuracy. When the data consist of errors with a heavy tail, suggestive of the possibility of outliers, GMM and Eq2r0 both perform equally well. Averaging over the replicates was a way to reduce the effect of the outliers. Models accounting for the replicates (i.e. Eq2r1) performed poorly with high MCE. When the data are heavily skewed (Gamma), none of the methods performed well. In this case, an extension to a mixture model defined by gamma distributions would most likely result in better clusters (Lakshmi and

Fig. 2. Clustering results from bone healing microarray study for strain AJ mice on the control diet (8934 differentially expressed genes with 3 replicates and 9 time-points) with entropy penalized EM-algorithm Eq0r0 ($p = 2$, no REs). Each plot represents temporal gene-expression profiles clustered into the same group

Fig. 3. Clustering results from bone healing microarray study for strain AJ mice on the control diet (8934 differentially expressed genes with 3 replicates and 9 time-points) with entropy penalized EM-algorithm Eq2r0 ($p = 2$, $q = 2$, no replicate-specific RE). Each plot represents temporal gene-expression profiles clustered into the same group

Vaidyanathan, 2016). The performance of any model-based clustering method is dependent on how well the model fits to the data.

For large datasets, despite the high accuracy of predicted class labels, a major limitation of this method is that the computation time is quadratic in growth for datasets with a large number of genes. It would be of interest to develop a package in C++ to improve computational efficiency. When we applied the algorithm to the fracture healing study, we found nine distinct clusters that appeared to be well separated with distinct patterns of temporal expression. Future work will be conducted by performing a detailed

analysis on the functional or regulatory properties of genes in each cluster.

With the rise of big-data and availability of high-dimensional data, the use of a clustering algorithm that can handle different sources of variability would be important to be able to determine clustering patterns that may not be easily obtained otherwise. We have shown that using the methods proposed in this paper, we can dramatically improve the accuracy of predicted class labels for a large set of genes.

Funding

Microarray data analysis was supported with grants from the National Institute of Arthritis and Musculoskeletal and Skin Diseases RO1AR05974 to LCG and National Center for Advancing Translational Sciences (BU-CTS1) 1UL1TR001430.

Conflict of Interest: none declared.

References

- Aike, H.A.I. (1974) A new look at the statistical model identification. *IEEE Trans. Automatic Control*, **19**, 716–723.
- Bais, M. et al. (2009) Transcriptional analysis of fracture healing and the induction of embryonic stem cell-related genes. *PloS One*, **4**, e5393.
- Biernacki, C. et al. (2000) Assessing a mixture model for clustering with the integrated classification likelihood. *IEEE Trans. Pattern Anal. Mach. Intell.*, **22**, 719–725.
- Biernacki, C. et al. (2003) Choosing starting values for the EM algorithm for getting the highest likelihood in multivariate Gaussian mixture models. *Comput. Stat. Data Anal.*, **41**, 561–575.
- Celeux, G. et al. (2005) Mixture of linear mixed models for clustering gene expression profiles from repeated microarray experiments. *Stat. Model.*, **5**, 243–267.
- Chamroukhi, F. (2016) Unsupervised learning of regression mixture models with unknown number of components. *J. Stat. Comput. Simul.*, **86**, 2308–2334.
- Dempster, A.A.P. et al. (1977) Maximum likelihood from incomplete data via the EM Algorithm. *J. R. Stat. Soc.*, **39**, 1–38.

- Figueiredo, M.A.T. et al. (2002) Unsupervised learning of finite mixture models. *IEEE Trans. Pattern Anal. Mach. Intell.*, **24**, 381–396.
- Fonesca, J.R. and Cardoso, M. (2007) Mixture-model cluster analysis using information theoretical criteria. *Intell. Data Anal.*, **11**, 155–173.
- Fraley, C. et al. (2012) mclust Version 4 for R: normal mixture modeling for model-based clustering, classification, and density estimation. *Technical Report*.
- Gaffney, S. (2004) Probabilistic curve-aligned clustering and prediction with regression mixture models. Ph.D. thesis, University of California, Irvine.
- Gaffney, S. and Smyth, P. (1999) Trajectory clustering with mixtures of regression models. In: *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 63–72.
- Grimes, R. et al. (2011) The transcriptome of fracture healing defines mechanisms of coordination of skeletal and vascular development during endochondral bone formation. *J. Bone Min. Res.*, **26**, 2597–2609.
- Hartigan, J.A. and Wong, M.A. (1979) Algorithm AS 136 A K-means clustering algorithm. *J. R. Stat. Soc. Ser. C (Appl. Stat.)*, **28**, 100–108.
- Hubert, L. and Arabie, P. (1985) Comparing partitions. *J. Classif.*, **2**, 193–218.
- Jepsen, K.J. et al. (2008) Genetic variation in the patterns of skeletal progenitor cell differentiation and progression during endochondral bone formation affects the rate of fracture healing. *J. Bone Min. Res.*, **23**, 1204–1216.
- Lakshmi, R.V. and Vaidyanathan, V. (2016) Parameter estimation in gamma mixture model using normal-based approximation. *J. Stat. Theory Appl.*, **15**, 25–35.
- McLachlan, G.J. and Krishnan, T. (2008) *The EM Algorithm and Extensions*, vol. 382. John Wiley & Sons, Hoboken, New Jersey.
- McLachlan, G.J. and Peel, D. (2000) Assessing the number of components in mixture models. In: *Finite Mixture Models*, John Wiley & Sons, New York, pp. 175–220.
- Ng, S.-K. et al. (2006) A mixture model with random-effects components for clustering correlated gene-expression profiles. *Bioinformatics*, **22**, 1745–1752.
- Roeder, K. et al. (1997) Practical Bayesian density estimation using mixtures of normals. *J. Am. Stat. Assoc.*, **92**, 894–902.
- Sneath, P.H. and Sokal, R.R.S. (1973) *Numerical Taxonomy. The Principles and Practice of Numerical Classification*, 1st edn. W. H. Freeman and Co, San Francisco.
- Storey, J.D. et al. (2005) Significance analysis of time course microarray experiments. *Proc. Natl. Acad. Sci. USA*, **102**, 12837–12842.
- Wigner, N.A. et al. (2010) Acute phosphate restriction leads to impaired fracture healing and resistance to bmp-2. *J. Bone Min. Res.*, **25**, 724–733.
- Yang, M.-s. et al. (2012) A robust EM clustering algorithm for Gaussian mixture models. *Pattern Recogn.*, **45**, 3950–3961.