Mixture models with multiple levels, with application to the analysis of multi-factor gene expression data.

Rebecka Jörnsten*
Department of Statistics
Rutgers University
501 Hill Center
Piscataway, NJ 08854, USA

Sündüz Keleş
Department of Statistics,
Department of Biostatistics
and Medical Bioinformatics
University of Wisconsin-Madison
1300 University Avenue
Madison, WI 53706, USA

November 19, 2006

Abstract

Model-based clustering is a popular tool for summarizing high-dimensional data. With the number of high-throughput large-scale gene expression studies still on the rise, the need for effective data summarizing tools has never been greater. By grouping genes according to a common experimental expression profile, we can gain new insights into the biological pathways that steer biological processes of interest. Clustering of gene profiles can also assist in assigning functions to genes that have not yet been functionally annotated.

*Corresponding Author: rebecka@stat.rutgers.edu, telephone +1 732 445-3145, fax +1 732 445-3428

Model-based clustering has to-date primarily been applied in a "single-level" setting: that is, a gene profile is defined across all experimental factor levels, regardless of whether one or more factors are studied. In many experiments, where two or more experimental factors are considered simultaneously, this can lead to a very inefficient model representation. For example, consider a two-factor experiment with factors "time" and "cell-line". A set of genes may exhibit a similar time course expression profile for one cell-line, while exhibiting several different time course profiles for a second cell-line. In other words, some profiles may coincide *between clusters* for a subset of levels of an experimental factor (here: cell-line). In addition, the description of a cluster profile may be significantly simplified with an efficient *within-cluster* parametrization. A particular time-course pattern may be common to both cell-lines, and clusters that define these patterns need only be distinguished by one set of profile parameters (excluding time/cell-line interactions). Other clusters may exhibit a flat time-course profile for one or both cell-lines, which yet again requires only a subset of model parameters to describe.

If a single-level model and full parametrization is enforced, it is possible that we both overfit and underfit the data; we may overfit by assigning an unnecessary degree of complexity to some clusters; we may underfit the number of clusters since we spent our parameter budget inefficiently (e.g. on flat cluster profiles, or clusters that coincide for a subset of model parameters).

We propose a mixture model with multiple levels, $\mathcal{MIX}_{\mathcal{L}}$, that provides sparse representations both *within* and *between* cluster profiles. We explore various flexible parameterizations for the cluster profiles, and discuss how an efficient parametrization can greatly enhance the objective interpretability of the generated clusters. Interpretable cluster profiles can assist in detecting biologically relevant groups of genes that may be missed with a less efficient parametrization. We use our multi-level parametrization as a basis for mining a proliferating cell-line expression data for annotational context and regulatory motifs. We also investigate the performance of the multi-level clustering approach on several simulated data sets.

Keywords: Clustering, Gene Expression, Mixture Model, Model Selection, Profile EM

1 Introduction

Model-based clustering is frequently used to summarize complex high-dimensional gene expression data. The base model is usually Gaussian, though some alternatives have been explored to account for outliers that do not group with any other data objects (Banfield and Raftery; 1993). The multivariate Gaussian mixture allows for clusters of varying shape, orientation and volume (Fraley and Raftery; 2002, 2004; Raftery and Dean; 2006). Many non-parametric clustering methodologies and algorithms have also been proposed for the analyses of genomic data. Non-parametric approaches may seem to be more flexible, which is indeed a desirable property when little is known at the onset of the analysis. However, many of the most commonly used non-parametric schemes are in fact very restrictive in that cluster shapes or volumes restriction are implicitly defined by the cost function of the clustering algorithms (Jornsten; 2004). We consider k-means as an example (or any center-based allocation schemes like PAM and k-median (Kaufman and Rousseeuw; 1990; Jornsten et al.; 2002)). By making cluster assignment solely dependent on the cluster center, cluster shape is ignored. Thus, k-means tends to produce spherical and equal size clusters, and is thus more restrictive than a model-based clustering approach where the cluster covariances are parameterized.

In this paper we discuss how to generate more interpretable and efficient data representations using multivariate gaussian mixture models. We address the following limitations of the current forms of single-level model-based clustering; (1) Model-based clustering usually treats all experimental conditions interchangeably even in the case of multi-factor experiments; (2) The literature on subset model selection for model-based clustering has mainly focused on the problem of identifying the *dimensions* that are informative with respect to cluster separation (Law et al.; 2004; Raftery and Dean;

2006; Tadesse et al.; 2005; Hoff; 2006), not the sparsest parameterized representation of each cluster mean.

We propose a multi-level mixture modeling approach that generates interpretable clusters in multiple-factor experiments. In the simplest setting, the first level of the mixture model clusters on one particular experimental factor (e.g. "time"), whereas the second level clusters the levels of the experimental factor of interest (e.g. "cell-line"). We let $\mathbf{x}_g$ and $\mathbf{y}_g$ denote the measurements for different levels of the factor of interest for gene g. We will denote the total number of clusters at the 1st level by K, and the number of second-level (sub)clusters, within each 1st level cluster k, by L_k. Let R_g and Z_g be two gene specific indicators denoting the class label at the 1st and 2nd levels. Our model assumes that

$$Pr(\mathbf{x}_g, \mathbf{y}_g \mid R_g = k, Z_g = l) \sim MVN(\mu_{kl}, \mathbf{\Sigma}_{kl}),$$

where μ_{kl} and $\mathbf{\Sigma}_{kl}$ represent the mean and variance-covariance matrix of the lth second level cluster within the k first level cluster. In addition, we parameterize the (sub)cluster means as $\mu_{kl} = W\beta_{kl} = W(\beta_k, \beta_{l(k)})'$, where β_k denotes the top-level cluster specific parameters, and $\beta_{l(k)}$ the sub-level specific parameters, and W is the design matrix for the multi-factor experiment. We perform subset selection on the parameters, not the dimensions. Thus, we always utilize the complete data set to generate clusters, and obtain cluster means that are directly interpretable in terms of between-experimental factors, and within-experimental factor expression. We will discuss specific choices of parameterizations in Section 2. We refer to Figure 1 for an example. In Figure 1 , $\beta_k = (\alpha_1, \alpha_2, \alpha_3)$ parameterizes the expression profile of cell-line 1 over time points (t_1, t_2, t_3), and $\beta_{l(k)} = (\gamma_1, \gamma_2, \gamma_3)$ parameterizes the *differential* expression profiles of cell-line 2 compared with 1 over these same set of time points.

Figure 2 (a) illustrates one motivation for introducing a mixture model with multiple levels, $\mathcal{MIX}_{\mathcal{L}}$: In some multi-factor experiments, certain effects may dominate the cluster models. Let's say that cell-line 1 is associated with larger differential expression

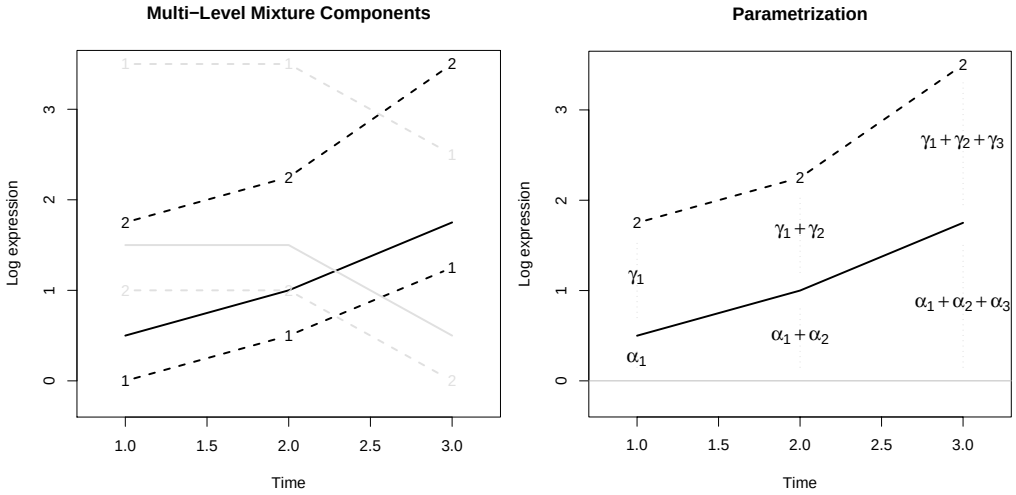

Figure 1: (a) An illustration of a Mixture Model with 2 levels; Solid line (black and gray): Two 1st level clusters for cell-line 1. The two sets of dashed lines (black and gray) represent the corresponding sub-clusters (level 2) for cell-line 2. Thus, here $K = 2$, and $L_1 = 2$, $L_2 = 2$. (b) The "Dynamic DE (differential expression) parametrization". The α parameters model the time course expression profile for cell-line 1, whereas the γ parameters model the time course of *cell-line differential* expression.

effects across time, and that these effects dominate the effects seen in cell-line 2. By treating time and cell-line as a single factor, we may fail to detect the more subtle patterns in cell-line 2. In addition, if some genes exhibit differential expression profiles in cell-line 2 but not in cell-line 1, a single-level mixture model is a very inefficient parametrization, over-fitting the cluster profile models for the cell-line 1 data.

In addition to accounting for dominating effect sizes, multi-level clustering models have much more general applicability. In experiments involving multiple species, or studies of gene expression in response to different treatment dosages, it is of interest to focus particularly on differential effects across levels of an experimental factor of interest (e.g. species, dose).

A few other schemes with a multi-level flavor have been proposed. Li (Li; 2005) introduced a layered mixture modeling approach to allow for more flexible within-cluster structures. Akin to MDA (Hastie and Tibshirani; 1996) for classification, each cluster (class) is assumed to come from a mixture of normals, and can thus incorporate more complex cluster (class) shapes. The number of clusters is assumed known, and

clusters do not share any mixture components with other clusters. Our multi-level mixture model differs from Li's approach in that an unknown number of clusters may share components and model parameters, and that the levels of the mixture relate to the experimental factors. Yuan and Kendziorski (Yuan and C.Kendziorski; 2006) recently proposed a multi-level approach to gene clustering and detection of differential expression. Each cluster is assumed to be generated from a mixture of differential expression patterns (over-expressed, under-expressed, and no differential expression). An empirical Bayes strategy is adopted to fit the model. The motivation is that the clustering induces a regularization of the gene effect estimates, and thus power of detection of differential expression is increased. Our multi-level approach allows for a more flexible parametrization of the cluster means across multiple experimental conditions. We identify differential expression patterns both within and between the experimental factors through model subset selection.

The paper is structured as follows. In section 2 we introduce our multi-level mixture model, and the Profile Expectation-Maximization algorithm we derive to fit the model. We describe our approach to parameterized subset selection, and validation of the number of clusters. We apply $\mathcal{MIX}_{\mathcal{L}}$ to a multi-factor gene expression data set of proliferating cell lines in section 3. We mine the clustering outcome for regulatory motifs and discuss the biological relevance. In section 4, we illustrate the strengths our approach on several simulated data sets. We conclude this paper with a discussion.

2 The $\mathcal{MIX}_{\mathcal{L}}$ model.

We begin by briefly reviewing the "single-level" model-based clustering method, and fix the notation for the subsequent discussion. We assume that for each data object $g \in \{1, \cdots, G\}$ we observe a feature vector $\mathbf{x}_g$. We denote cluster membership indicators by R_g, where $R_g = k$ if object g belongs to cluster k. We further assume that the objects are independent given the cluster memberships $R_g = k$. The most common approach

is to assume that each cluster or mixture component follows a Gaussian distribution:

$$Pr(\mathbf{x}_g \mid R_g = k) \sim MVN(\mu_k, \mathbf{\Sigma}_k).$$

The marginal distribution of the data is

$$Pr(\mathbf{x}_g) = \sum_{k=1}^{K} \pi_k \psi(\mathbf{x}_g; \mu_k, \mathbf{\Sigma}_k),$$

where ψ is the multivariate normal density function. The parameters $\theta = \{\theta_k = (\pi_k, \mu_k, \mathbf{\Sigma}_k), \forall k\}$ are commonly estimated with the Expectation-Maximization algorithm (Dempster et al.; 1977). Convergence is usually quite fast. To avoid settling for a local solution, the EM steps are usually re-run from multiple starting points (McLachlan and Peel.; 2000). It is well established that unconstrained fitting of the K component mixture model can suffer from singularity problems where a single object can form its own cluster with a degenerate distribution. To prevent this, it is recommended that one employs constrained optimization with respect to the covariance parameters (e.g. Celeux and Govaert (1993)), or that one regularizes the fit by shrinking toward the global covariance (Fraley and Raftery; 2004).

2.1 A multi-level parametrization for model-based clustering

For the sake of presentation, we will consider an experiment where there are two populations (e.g. cell lines) of interest, and samples from both of these populations are collected across T time points. Let $\mathbf{x}_g$ denote the observations across T time points for gene g in cell line 1, and similarly $\mathbf{y}_g$ in cell line 2. The hierarchy we will consider treats the cell line as the factor interest. We will denote the total number of clusters at the 1st level by K, and the number of clusters within each 1st level cluster k by L_k. Let R_g and Z_g be two gene specific indicators denoting the class label at the 1st and

2nd level. Our model assumes that

$$Pr(\mathbf{x}_g, \mathbf{y}_g \mid R_g = k, Z_g = l) \sim MVN(\mu_{kl}, \mathbf{\Sigma}_{kl}),$$

where μ_{kl} and $\mathbf{\Sigma}_{kl}$ represent the mean and variance-covariance matrix of the l-th second level cluster within the k first level cluster. Here, the first T components of the μ_{kl} vector correspond to the mean levels of $\mathbf{x}_g$, and will be also referred to as μ_k. The last T components correspond to the mean levels of $\mathbf{y}_g$, and will be referred to as $\mu_{l(k)}$. This multi-level framework allows for various interpretable parameterizations at each level.

We will be utilizing a data set on proliferating stem cell-lines to demonstrate the $\mathcal{MIX}_{\mathcal{L}}$ method. Our task is to identify sets of genes that are differentially regulated during neurogenesis and gliogenesis, as indicated by different expression levels in two, divergent neural stem cell (NSC) clones. Upon the withdrawal of a growth factor (FGF) from the medium, one clone (L2.3) becomes predominately glial-like (expressing glial markers GFAP, GalC). The other (L2.2) differentiates primarily into cells expressing neuronal markers (TuJ1) (Goff et al.; 2006). At the starting point (time t=0), the cell-lines are virtually indistinguishable, and are believed to exist in a state of "pre-conditioning" or "pre-programming". Thus, sets of neuron-specific and glia-specific genes are active, and will determine the cell-fate of the clones. The two stem cell-lines were observed over the course of three days, for a total of $T = 3$ sample points ($t = (0, 1, 3)$ days). We denote gene expression in the glial-like population (L2.3) by $\mathbf{x}$, and in the neuron-like (L2.2) population by $\mathbf{y}$. Among the several scientific questions of interest given to us by the biologists (Goff et al. (2006)) were; (a) How do the time course profiles of the glial-like (L2.3) and neuron-like (L2.2) cell-lines differ?; (b) Are there sets of genes for which the expression converges(diverges) between the glial-like and neuron-like cell populations?; (c) How dominant is the "pre-programming" effect? To address these questions, we consider the following three parameterizations:

Parametrization I. *Mean differential expression.*

$$\begin{aligned}\mu_k &= (\mu_{k1}, \mu_{k2}, \mu_{k3})' \\ \mu_{kl} &= (\mu_{k1}, \mu_{k2}, \mu_{k3}, \mu_{k1} + \Delta_{k1}, \mu_{k2} + \Delta_{k2}, \mu_{k3} + \Delta_{k3})'\end{aligned}$$

Here, the main scientific question addressed is the differential expression between the cell-lines, at any given time point.

Parametrization II. *Dynamical differential expression.*

$$\begin{aligned}\mu_k &= (\alpha_{k1}, \alpha_{k1} + \alpha_{k2}, \alpha_{k1} + \alpha_{k2} + \alpha_{k3})' \\ \mu_{kl} &= (\alpha_{k1}, \alpha_{k1} + \alpha_{k2}, \alpha_{k1} + \alpha_{k2} + \alpha_{k3}, \alpha_{k1} + \gamma_{kl1}, \alpha_{k1} + \alpha_{k2} + \gamma_{kl1} + \gamma_{kl2}, \\ &\quad \alpha_{k1} + \alpha_{k2} + \alpha_{k3} + \gamma_{kl1} + \gamma_{kl2} + \gamma_{kl3})'\end{aligned}$$

In the second parametrization, the time course profile of the glial-like population is modeled directly, and e.g. flat time profiles are efficiently represented. The γ-vector represents the *time-course* of cell-line differential expression (e.g. parallel, divergent or convergent).

Parametrization III. *Pre-programming differential expression.*

$$\begin{aligned}\mu_k &= (\alpha_{k1}, \alpha_{k1} + \alpha_{k2}, \alpha_{k1} + \alpha_{k2} + \alpha_{k3})' \\ \mu_{kl} &= (\alpha_{k1}, \alpha_{k1} + \alpha_{k2}, \alpha_{k1} + \alpha_{k2} + \alpha_{k3}, \alpha_{k1} + \gamma_{kl1}, \alpha_{k1} + \gamma_{kl1} + \gamma_{kl2}, \\ &\quad \alpha_{k1} + \gamma_{kl1} + \gamma_{kl2} + \gamma_{kl3})'\end{aligned}$$

The third parametrization efficiently models each time course-profile, and a main differential cell-line effect for time point 0.

Other data sets and experimental structures may require a different set of parameterizations. Ultimately, the choice of parametrization should depend on the biological

context, and the scientific questions of interest.

In all parameterizations, the variance-covariance matrix $\mathbf{\Sigma}_{kl}$ also includes parameters specific to the levels of the model:

$$\mathbf{\Sigma}_{kl} = \begin{bmatrix} \mathbf{\Sigma}_k^X & \mathbf{\Sigma}_{kl}^{XY} \\ \mathbf{\Sigma}_{kl}^{YX} & \mathbf{\Sigma}_{kl}^{Y} \end{bmatrix}.$$

The variance-covariance structure allows for dependencies between gene expression measurements at all time points and all levels. We further assume that, conditional on the multi-level cluster assignments, the genes are independent of each other. Therefore, we have the following complete data likelihood:

$$\begin{aligned} Pr(\mathbf{X}, \mathbf{Y}, \mathbf{R}, \mathbf{Z} \mid \mathbf{\Psi}) &= \prod_{g=1}^{G} Pr(\mathbf{x}_g, \mathbf{y}_g, R_g, Z_g \mid \mathbf{\Psi}) \\ &= \prod_{g=1}^{G} \prod_{k=1}^{K} \prod_{l=1}^{L} [Pr(\mathbf{x}_g, \mathbf{y}_g \mid R_g = k, Z_g = l)\pi_{kl}]^{I(R_g=k, Z_g=l)}, \end{aligned}$$

where $\mathbf{\Psi}$ represents the overall parameter set of the model. Due to the multi-level parametrization and the general variance-covariance structure, the standard update schemes for the Expectation-Maximization algorithm of the mixture models are not easily adapted. Therefore, we develop a Profile Expectation-Maximization (PEM) algorithm. In the next section, we outline this algorithm in detail. The algorithm relies on working with the factorization of the likelihood into the likelihood of the 1st level of the hierarchy, and the conditional likelihood of the 2nd level of the hierarchy given the first level. Additionally, each component of the factorized likelihood is maximized by profiling of the corresponding expected complete data log likelihood.

2.2 Profile Expectation-Maximization (PEM) algorithm for fitting the multi-level mixture model

We now describe the profile Expectation-Maximization algorithm for fixed K and L_k, $k \in \{1, \cdots, K\}$. In what follows, r refers to r-th EM iteration and we suppress the

dependence of the updates on r to ease the notation.

Initial values. The algorithm requires initial values of π_{kl} and μ_{kl} and $\mathbf{\Sigma}_{kl}$. We will discuss the initialization step at the end of this section.

E-step. This step is a regular E-step in fitting mixture of multivariate normals. We have posterior class probabilities given by

$$\begin{aligned} \eta_{gkl}^{(r)} \equiv \hat{\eta}_{gkl} &= Pr(R_g = k, Z_g = l \mid \mathbf{x}_g, \mathbf{y}_g, \psi^{(r-1)}) \\ &= \frac{MVN(\mathbf{x}_g, \mathbf{y}_g \mid \mu_{kl}^{(r-1)}, \mathbf{\Sigma}_{kl}^{(r-1)})\pi_{kl}^{(r-1)}}{Pr(\mathbf{x}_g, \mathbf{y}_g \mid \psi^{(r-1)})}. \end{aligned}$$

M-step. In the M-step, we are dealing with the following maximization problem

$$\sum_{g=1}^{G}\sum_{k=1}^{K}\sum_{l=1}^{L_k}\left(-\frac{1}{2}\hat{\eta}_{gkl}(\mathbf{u}_g - W\beta_{kl})'\mathbf{\Sigma}_{kl}^{-1}(\mathbf{u}_g - W\beta_{kl}) - \frac{1}{2}\hat{\eta}_{gkl}\log|\mathbf{\Sigma}_{kl}|\right), \tag{1}$$

where W represents the design matrix corresponding to the parametrization and $\mathbf{u}_g$ is the combined vector of $\mathbf{x}_g$ and $\mathbf{y}_g$. For example, for parametrization I:

$$W_I = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

with $\beta_{kl} = (\mu_{k1}, \mu_{k2}, \mu_{k3}, \Delta_{kl1}, \Delta_{kl2}, \Delta_{kl3})$, and for parameterizations II and III:

$$W_{II} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

$$W_{III} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 1 \end{pmatrix}$$

with $\beta_{kl} = (\alpha_{k1}, \alpha_{k2}, \alpha_{k3}, \gamma_{kl1}, \gamma_{kl2}, \gamma_{kl3})$.

The main reason for a nonstandard mixture model M-step is due to the cross-talk between the two levels. The first part of the parameter vector β_{kl} is the same for all l, and similarly the left upper diagonal block $\mathbf{\Sigma}_k^X$ of $\mathbf{\Sigma}_{kl}$ is common to all l. Hence, the corresponding estimates need to pool information across all second level clusters of the kth 1st level cluster. We use a regularized profiling method for maximizing the expected complete data log likelihood given in equation 1. Our general iterative scheme is to factorize the joint likelihood of $\mathbf{x}_g$ and $\mathbf{y}_g$ as the product of marginal likelihood of $\mathbf{x}_g$, and the conditional likelihood of $\mathbf{y}_g$ given $\mathbf{x}_g$. We first maximize the marginal likelihood of $\mathbf{x}_g$ by profiling. Subsequently, given the estimates μ_k and $\mathbf{\Sigma}_k$, we maximize the conditional likelihood of $\mathbf{y}_g$ given $\mathbf{x}_g$, again by profiling over the mean and the variance-covariance matrix. Next, we outline these two steps in more detail.

1. **M-step-Aggregate.** Compute gene specific membership for level 1 by aggregat-

ing $\hat{\eta}_{gkl}$.

$$\hat{\tau}_{gk} = \sum_{l=1}^{L_k} \hat{\eta}_{gkl}.$$

2. **M-step-Profile-1.** This step concerns the profiling of the multivariate normal density for $\mathbf{x}_g$.

   (a) Considering part of the expected complete data likelihood that involves the marginal distribution of $\mathbf{x}_g$, we have

   $$\sum_{g=1}^{G}\sum_{k=1}^{K}\sum_{l=1}^{L_k}\left(-\frac{1}{2}\hat{\eta}_{gkl}(\mathbf{x}_g - W_K\beta_k)'\mathbf{\Sigma}_k^{-1}(\mathbf{x}_g - W_K\beta_k) - \frac{1}{2}\hat{\eta}_{gkl}\log|\mathbf{\Sigma}_k^X|\right) = \quad (2)$$
   $$\sum_{g=1}^{G}\sum_{k=1}^{K}\left(-\frac{1}{2}\hat{\tau}_{gk}(\mathbf{x}_g - W_K\beta_k)'\mathbf{\Sigma}_k^{-1}(\mathbf{x}_g - W_K\beta_k) - \frac{1}{2}\hat{\tau}_{gk}\log|\mathbf{\Sigma}_k^X|\right),$$

   where W_K is the upper left diagonal block of design matrix W (corresponding to the 1st level data).

   (b) Given μ_k, we profile with respect to $\mathbf{\Sigma}_k^X$ and get

   $$\mathbf{\Sigma}_k^{X^{(r)}} = \frac{\sum_{g=1}^{G}\hat{\tau}_{gk}(\mathbf{x}_g - \mu_k^{(r)})(\mathbf{x}_g - \mu_k^{(r)})'}{\sum_{g=1}^{G}\hat{\tau}_{gk}}.$$

   Then, a regularized version of $\mathbf{\Sigma}_k^X$ is obtained by

   $$\tilde{\mathbf{\Sigma}}_k^{X^{(r)}} = \frac{\Delta_p^X(\nu) + \mathbf{\Sigma}_k^{X^{(r)}} n_k}{\nu + n_k},$$

   where $n_k = \sum_{g=1}^{G}\hat{\tau}_{gk}$, and $\Delta_p^X = \frac{\sum_{g=1}^{G}(\mathbf{x}_g - \bar{x_g})(\mathbf{x}_g - \bar{x_g}))'}{GK^{2/T}}$. The choice of scale parameter, ν, is discussed in section 2.4 (see also Fraley and Raftery (2004)).

   (c) Holding $\mathbf{\Sigma}_k^X$ fixed, the maximizer over β_k can be obtained via weighted least squares. In fact, with some simple algebra, one can show that estimate of β_k can be obtained by first performing a weighted least squares fit of the form

   $$x_g = W_K\beta_{gk} + \epsilon, \quad \text{where} \quad cov(\epsilon) = \mathbf{\Sigma}_k^X,$$

and then taking a weighted average of the estimates of β_{gk} as

$$\beta_k^{(r)} \equiv \hat{\beta}_k = \frac{\sum_{g=1}^{G} \hat{\tau}_{gk} \hat{\beta}_{gk}}{\sum_{g=1}^{G} \hat{\tau}_{gk}}.$$

Finally, we update μ_k as $\mu_k^{(r)} = W_K \hat{\beta}_k$.

3. **M-step-Profile-2.** Next, we consider profiling the conditional distribution of $\mathbf{y}_g$ given $\mathbf{x}_g$. The expected complete data likelihood that involves the conditional distribution of $\mathbf{y}_g$ given $\mathbf{x}_g$ is given by

$$\sum_{g=1}^{G} \sum_{k=1}^{K} \sum_{l=1}^{L_k} \left(-\frac{1}{2} \eta_{gkl} (\mathbf{y}_g - \mu_{kl}^{Y|X})' \mathbf{\Sigma}_{kl}^{Y|X^{-1}} (\mathbf{y}_g - \mu_{kl}^{Y|X}) - \frac{1}{2} \eta_{gkl} \log |\mathbf{\Sigma}_k^{Y|X}| \right),$$

where

$$\begin{aligned} \mu_{kl}^{Y|X} &= \mathbf{\Sigma}_{kl}^{XY} (\mathbf{\Sigma}_k^X)^{-1} (\mathbf{x}_g - \mu_k), \\ \mathbf{\Sigma}_{kl}^{Y|X} &= \mathbf{\Sigma}_{kl}^{Y} - \mathbf{\Sigma}_{kl}^{YX} (\mathbf{\Sigma}_k^X)^{-1} \mathbf{\Sigma}_{kl}^{XY}. \end{aligned}$$

(a) The second profiling step starts with updating $\mathbf{\Sigma}_{kl}^{YX}$and $\mathbf{\Sigma}_{kl}^{Y}$ as follows.

$$\begin{aligned} \mathbf{\Sigma}_{kl}^{YX^{(r)}} &= \frac{\sum_{g=}^{G} \hat{\eta}_{gkl} (\mathbf{y}_g - \mu_{kl}^{Y^{(r)}})(\mathbf{x}_g - \mu_k^{X^{(r)}})'}{\sum_{g=1}^{G} \hat{\eta}_{gkl}}, \\ \mathbf{\Sigma}_{kl}^{Y^{(r)}} &= \frac{\sum_{g=}^{G} \hat{\eta}_{gkl} (\mathbf{y}_g - \mu_{kl}^{Y^{(r)}})(\mathbf{y}_g - \mu_{kl}^{Y^{(r)}})'}{\sum_{g=1}^{G} \hat{\eta}_{gkl}}. \end{aligned}$$

The regularized versions of these covariance estimates are

$$\begin{aligned} \tilde{\mathbf{\Sigma}}_{kl}^{YX^{(r)}} &= \frac{\Delta_p^{YX}(\nu) + \mathbf{\Sigma}_{kl}^{YX^{(r)}} n_{kl}}{\nu + n_{kl}}, \\ \tilde{\mathbf{\Sigma}}_{kl}^{Y^{(r)}} &= \frac{\Delta_p^{Y}(\nu) + \mathbf{\Sigma}_{kl}^{Y^{(r)}} n_{kl}}{\nu + n_{kl}}, \end{aligned}$$

where

$$\begin{aligned}\Delta_p^Y &= \frac{\sum_{i=1}^G (\mathbf{y}_g - \bar{\mathbf{y}})(\mathbf{y}_g - \bar{\mathbf{y}})'}{\sum_{k=1}^K {L_k}^{2/d}}, \\ \Delta_p^{YX} &= \frac{\sum_{i=1}^G (\mathbf{y}_g - \bar{\mathbf{y}})(\mathbf{x}_g - \bar{\mathbf{x}})'}{\sum_{k=1}^K {L_k}^{2/d}}.\end{aligned}$$

Then, the conditional mean of $\mathbf{y}_g$ and the covariance matrix are updated as follows:

$$\begin{aligned}\mu_{kl}^{Y|X^{(r)}} &= \mathbf{\Sigma}_{kl}^{XY^{(r)}} (\mathbf{\Sigma}_k^{X^{(r)}})^{-1} (\mathbf{x}_g - \mu_k^{(r)}), \\ \mathbf{\Sigma}_{kl}^{Y|X^{(r)}} &= \mathbf{\Sigma}_{kl}^{Y^{(r)}} - \mathbf{\Sigma}_{kl}^{YX^{(r)}} (\mathbf{\Sigma}_k^{X^{(r)}})^{-1} \mathbf{\Sigma}_{kl}^{XY^{(r)}}.\end{aligned}$$

(b) Similar to the **M-step-Profile-1** step above, for fixed $\mathbf{\Sigma}_{kl}^{Y|X}$, we have a weighted least squares formulation given by

$$\mathbf{y}_g^* = W_L \beta_{gl(k)} + \epsilon, \quad cov(\epsilon) = \mathbf{\Sigma}_{kl}^{Y|X},$$

where $\mathbf{y}_g^* = \mathbf{y}_g - \mu_{kl}^{Y|X^{(r)}} - W_{LK} \hat{\beta}_k^{(r)}$ and W_L represents the lower diagonal block of the W matrix corresponding to L-level parameters whereas W_{LK} represents the lower off-diagonal block of the W matrix.

We then have

$$\hat{\beta}_{l(k)} = \frac{\sum_{g=1}^G \hat{\eta}_{gkl} \hat{\beta}_{gl(k)}}{\sum_{g=1}^G \hat{\eta}_{gkl}},$$

and set $\hat{\beta}_{kl} = (\hat{\beta}_k, \hat{\beta}_{l(k)})$ and $\mu_{kl}^{(r)} = W \hat{\beta}_{kl}$.

Next, we summarize a single step of the PEM algorithm concisely.

Profile EM algorithm

1. **E-step.** Compute $\hat{\eta}_{gkl}$, $g = 1, \cdots, G$, $k = 1, \cdots, K$ and $l = 1, \cdots, L_k$.
2. **M-step.**
   (a) Update $\hat{\pi}_{kl}$, $k = 1, \cdots, K$ and $l = 1, \cdots, L_k$.

(b) **M-step-Aggregate.** Compute $\hat{\tau}_{gk} = \sum_{l=1}^{L_k} \hat{\eta}_{gkl}$.

(c) **M-step-Profile-1.**

i. Update $\mathbf{\Sigma}_k^X$.

ii. Update μ_k by reestimating $\hat{\beta}_k$ with the corresponding weighted least squares fit.

iii. Iterate (i) and (ii) till convergence.

(d) **M-step-Profile-2.**

i. Update $\mathbf{\Sigma}_{kl}^{XY}$, $\mathbf{\Sigma}_{kl}^{YX}$, $\mathbf{\Sigma}_{kl}^{Y}$ and $\mu_{kl}^{Y|X}$.

ii. Update μ_{kl} by reestimating $\beta_{l(k)}$ with the corresponding weighted least squares fit and by setting $\hat{\beta}_{kl} = (\hat{\beta}_k, \hat{\beta}_{l(k)})'$.

iii. Iterate (i) and (ii) till convergence.

Although the profiling steps could in principle benefit from internal iterations ((iii) above), we noticed in our applications that, in general, it is advantageous to spend the computing time on the outer EM iterations.

2.3 Model selection

Model selection in multi-level model-based clustering pertains to two components; (1) selecting the appropriate parametrization for each cluster $\{k, l\}$; (2) selecting the number of clusters K, and $\mathbf{L}_K = \{L_k, k = 1, \cdots, K\}$.

2.3.1 Cluster parameterizations and subset selection

Let us first consider the case with K and $\mathbf{L}_K$ fixed. We want to select the sparsest representation of each cluster mean. This will enable us to better interpret the meaning of each cluster. For example, is a particular cluster model representing (i) a static cell-line difference, or (ii) a dynamic one, and if so for which time-points do the cell-lines really differ?

Recently, several papers have appeared on the topic of variable selection for model based clustering. These papers focus on the selection of a subset of variables, or dimensions of the feature vector, that can discriminate between cluster components (e.g., Friedman and Meulman (2002), Law et al. (2004), Raftery and Dean (2006), Hoff (2006), Tadesse et al. (2005)).

Raftery et al. (Raftery and Dean; 2006) proposed an iterative algorithm, considering deletions or additions to the set of discriminative variables. Consider the addition of a set of variables. The two models that are compared are; (1) a cluster mixture model for the new set of variables (original set and the set under consideration), and a cluster independent model of the excluded variables, and (2) a cluster model for the original set, with a cluster independent model for the set under consideration and the excluded variables. The decision to accept a new set of variables is made using Bayes factors.

Hoff (Hoff; 2006) models the cluster means with cluster specific contrasts. Let us consider a d-dimensional data set with global mean $\mathbf{u}$ (d-dimensional) and covariance Σ. At the cluster level, we define parameters $\mathbf{u}_k = \mu + \delta^k$, where δ_k represents a set of contrasts between the global mean and the cluster mean. Hoff considers the case where only a subset of the d-dimensional vector δ_k are non-zero, and that this subset may vary across clusters. The model is fit via a hierarchical Bayesian scheme with priors on the cluster specific subsets of non-zero contrasts.

In our parametrization of the cluster means, as outlined in the section below, we deviate from the above approaches. Our parametrization, and the corresponding sparsest representation we select, allows for cluster specific descriptions of contrasts between variables *within* a cluster, as well as *between* clusters. We model all dimensions within the clustering model. However, for each cluster we allow for only a subset of *parameters* to be non-zero. The subset of coefficients that are set to zero do not necessarily correspond to a dimension that is irrelevant for clustering. Take as an example our first parametrization $(\mu, \mathbf{\Delta})$. If for a cluster k, the subset of parameters relating to μ are set to 0, then this dimension is unrelated to the clustering (assuming the data has

been centered prior to clustering). If however, a subset of parameters related to Δ is set to 0, this implies that the cluster consists of a set of genes for which there is no cell-line difference.

How do we then perform subset selection within each cluster model? Clearly, a full combinatorial search of all possible subsets is not feasible. For each combination of subset models, the EM algorithm has to be re-run to adapt to the reduced complexity of some of the clusters. Object posterior probabilities are affected by the cluster specific models.

We take a backward selection approach to selecting the optimal subset models. We begin with the full model at each node $\{k, l\}$. We then visit each node, one at a time, and threshold the posterior probabilities η_{gkl} to obtain a cluster specific data set of size n_{kl} (or n_k for an internal node k). We perform backward selection at an internal node k using only the K-level data. We formulate the model selection as a generalized linear regression problem, where $x_g = W_K\beta_k + \epsilon$, $\epsilon \sim N(0, \Sigma_K^X)$. We hold Σ_K fixed during the model selection, and the estimated covariance matrix is used in the weighted least squares fit. We use the local BIC to select the optimal cluster specific model. After backward selection we thus obtain a sparse solution β_k^* for each internal node. We then re-run the EM steps with the sparse restrictions on β (i.e. using a subset of the columns of matrices W_K for each cluster k). Thus we obtain an updated allocation between all $\{k, l\}$ nodes given the selected subset model class.

To perform model selection at the $\{k, l\}$ leaf-nodes we use the profile likelihood, as was done in the corresponding M-step of the fitting algorithm. For each leaf node $\{k, l\}$ we compute the conditional mean $\mu_{l(k)}$ and covariance $\Sigma_{k,l}^{Y|X}$. We can write the profile likelihood in terms of the L-specific parameters only ($\beta_{l(k)}$). We perform backward selection in a generalized linear regression problem; $y_g = W_L\beta_{l(k)} + \epsilon_L$, $\epsilon_L \sim N(0, \Sigma_{k,l}^{Y|X})$. We obtain the optimal sparse solution $\beta_{l(k)}^*$. We then re-run the EM steps with the sparse restrictions on $\beta_{l(k)}$ (a subset of columns of W_L for each sub-cluster $l(k)$). We thus obtain an updated allocation among the internal nodes and leaf nodes.

Finally, to reduce the impact of such a greedy and directed search, we re-run the whole selection strategy from the most recent allocation, starting yet again from the full model and searching backwards. In practice, we found that iterations of the subset selection algorithm rarely produced a different final result.

We outline the subset selection algorithm here:

I Initialize with the full model at each node z, where z is one from the set of internal ($k = 1, \cdots, K$) or leaf-nodes ($\{k, l\}, k = 1, \cdots, K, l = 1, \cdots, L_k$).

Set the current design matrix of each node z to the full W; $W_K(k)$ for the internal nodes, $W_L(k, l)$ for the leaf-nodes.

(The number of columns of a design matrix, $col(W(z))$, corresponds to the number of non-zero parameters at node z.)

Run the EM-algorithm.

II **(a)** Visit each internal node k, and perform a hard threshold operation on τ_{gk} to obtain the node specific data.

(b) If $W(k)$ is empty, go to the next node k.

Otherwise, perform backward selection for the weighted least squares fit at node k. Obtain the sparse solution β_k^* via the local BIC, and update the current design matrix at node k to $W_K(k) = W_K^*(k)$ (i.e. drop the columns that correspond to $\beta_k^* = 0$).

(c) Re-run the EM algorithm with the updated $W_K(k)$ constraints.

III **(a)** Visit each leaf node $\{k, l\}$, and perform a hard threshold operation on η_{gkl} to obtain the node specific data.

(b) If $W_L(l(k))$ is empty, go to the next node $\{k, l\}$.

Otherwise, perform backward selection for the weighted least squares fit at node $\{k, l\}$ using the profile likelihood. Obtain the sparse solution $\beta_{l(k)}^*$ via the local BIC, and update the current design matrix at node $\{k, l\}$ to

$W_L(l(k)) = W_L^*(l(k))$ (i.e. drop the columns that correspond to $\beta^*_{l(k)} = 0$).

(c) Re-run the EM algorithm with the $W_K(k)$ and updated $W_L(l(k))$ constraints.

IV Go to I and iterate until convergence.

2.3.2 Selecting the number of clusters.

The selection of the number of clusters is usually approached as a complexity allocation problem using criteria such as BIC, CIC or MDL (e.g. Fraley and Raftery (2002), Raftery and Dean (2006)). Recently, Zhu and Zhang (2004) developed a general statistical hypothesis testing formulation to select the number of clusters. Here we take the complexity allocation route, using BIC to select the number of clusters. Let us consider a multi-level parametrization where the dimensionality of the data vectors at level K is $Dim(K)$, and at level L $Dim(L)$. We denote the model coefficients at the $K - level$ by $\beta_k, k = \{1, \cdots, K\}$, and the model coefficients at the level by $\beta_l(k), l = \{1, \cdots, L_k\}$ for all $k = \{1, \cdots, K\}$. In the previous section we considered subset model selection for each node $\{k, l\}$ of the multi-level clustering. Thus, the number of non-zero coefficients $\beta_k \neq 0$ may be less than $DimK$, and similarly for $\beta_{l(k)}$. We denote the number of non-zero coefficients at each node $\{k, l\}$ by $(dim(\beta_k), \dim(\beta_{l(k)}))$ respectively. We gather all parameters of a multi-level fit into a set $\Theta(K, \mathbf{L}_K)$, where

$$\Theta(K, \mathbf{L}_K) = \{\pi_{kl}, \beta_k, \beta l(k), \Sigma_{kl}, \forall k = \{1, \cdots, K\}, l = \{1, \cdots, L_k\}\}.$$

Then the total model complexity is given by

$$p(\Theta(K, \mathbf{L}_K)) = \left[\sum_{k=1}^{K} \left(dim(\beta_k) + \sum_{l=1}^{L_k} dim(\beta_{l(k)}) \right)\right]_{(1)} +$$

$$+ \left[\frac{K Dim(K)(Dim(K) - 1)}{2}\right]_{(2)} + \left[(\sum_{k=1}^{K} L_k) - 1\right]_{(3)}$$

$$\left[\left(\sum_{k=1}^{K} L_k\right)\left(Dim(K)Dim(L) + \frac{Diml(L)(Dim(L)-1)}{2}\right)\right]_{(4)},$$

where term (1) is the number of mean parameters estimated at the K and L levels, term (2) is the K-level covariance estimates, term (4) is the L-level covariance estimates and cross-covariance estimates between the K and L levels, and term (3) is the number of estimated cluster proportions. For each given K and $\mathbf{L}_K$ we can compute the log-likelihood:

$$l(\Theta(K, \mathbf{L}_K)) = \sum_{g=1}^{G} \log\left(\sum_{k=1}^{K}\sum_{l=1}^{L_k} \pi_{kl}\phi((\mathbf{x}_g, \mathbf{y}_g); W\beta_{kl}, \Sigma_{kl})\right).$$

We then compute the BIC value as

$$BIC(K, \mathbf{L}_K) = -2l(\Theta(K, \mathbf{L}_K)) + p(\Theta(K, \mathbf{L}_K))\log(G).$$

We explored several different search strategies for identifying the optimal multi-level model. The best performance was obtained using a backward search. In the flow-chart below, M refers to the total number of clusters ($M = \sum_k L_k$).

I Initialize with the null model $M = 1, L_1 = 0$ and set the BIC to an arbitrarily large value.

II Set $M = M + 1$.

(a) Outer loop

- Set $K = M$ and $\mathbf{L}_K = \{L_k = 1, \forall k = \{1, \cdots, M\}\}$.

  Run the EM algorithm.

  Record the corresponding BIC value: $BIC(new)$.

  Go to Inner Loop **II-b**.

(b) Inner Loop

- Set K = K - 1

For $b = \{1, \cdots, B\}$

- group the M 1st level parameters from the single-level clustering (II-a): $(\mu_{\mathbf{k}}, \Sigma_k)$ into K groups. The corresponding grouping defines the set $\mathbf{L}_K^b(new) = \{L_k^b, k = 1, \cdots, K\}$.
- run the EM algorithm for K and $\mathbf{L}_K^b(new)$ and record $BIC^b(K)$.

• Set $b^* = argmin_b BIC^b(K)$, and set $BIC(K) = BIC^{b^*}(K)$. Retain the best multi-level clustering with K 1st level clusters and the corresponding grouping $\mathbf{L}_K(new) = \mathbf{L}_K^{b^*}(new)$.

(c) • If $BIC(K) \geq BIC(new)$ go to step **III** (the optimal number of sub-clusters has been exceeded).

• If $BIC(K) < BIC(new)$, accept the best multi-level model model K and the corresponding set $\mathbf{L}_K = \mathbf{L}_K(new)$, $BIC(new) = BIC(K)$. Go to Inner Loop step **II-b**.

III • If $BIC(new) \geq BIC$, STOP (the optimal number of clusters have been exceeded)

• If $BIC(new) < BIC$, set $BIC = BIC(new)$ and go to **II-a** (consider increasing the total number of clusters).

For both subset selection, and the selection of the number of clusters, we adopt greedy searches. While it is true that such schemes can converge to local optima, a fully exhaustive search is computationally prohibitive. A stochastic search may remedy the problem of local optima. We did not consider stochastic searches here, but do run the full algorithm several times while initiating from different starting values.

2.4 Computational details

2.4.1 Regularizing the cluster covariance estimates

In Fraley and Raftery (2004), a regularized estimate of the cluster covariances are introduces as

$$\tilde{\mathbf{\Sigma}}_k^{X^{(r)}} = \frac{\Delta_p^X(\nu_p + d + 2) + \mathbf{\Sigma}_k^{X^{(r)}} n_k}{\nu_p + d + 2 + n_k}.$$

The motivation for this regularization comes from assuming a conjugate inverse Wishart prior distribution with scale matrix Δ_0 and degrees of freedom ν_p for $\mathbf{\Sigma}_{kl}^X$. Here, Δ_0 is estimated by the plug-in estimator

$$\Delta_p^X = \frac{\sum_{i=1}^G (\mathbf{x}_g - \bar{\mathbf{x}})(\mathbf{x}_g - \bar{\mathbf{x}})'}{K^{2/d}},$$

where $\bar{\mathbf{x}}$ represents the componentwise mean vector over all the G genes. ν_p is chosen as $\max\{0, n_{min}\} + d + 2$, where d is the dimension of the data, and n_{min} can be interpreted as the number of observations with variance Δ_p^X that are added to the clustered data.

The scaled global covariance matrix is not always a good choice to shrink toward. Consider a clustering in two dimensions, where K clusters means lie on the 45 degree line, and the cluster covariance are aligned at 135 degrees (i.e. orthogonal to the line connecting the cluster means). The global covariance will be aligned with the 45 degree line. The weighted average between the Δ_p^X and Σ_k^X can thus produce a very different cluster shape, even for moderately large clusters. To reduce the impact of "over-regularizing" the covariance estimates we take a frequentist approach. We numerically test the regularized estimates

$$\Sigma_k^{X^{(r)}} = \frac{\Delta_p^X(\nu) + \mathbf{\Sigma}_k^{X^{(r)}} n_k}{\nu + n_k}.$$

with $\nu = 0$ for singularity problems. We increase ν gradually until the regularized estimate is functional. Although this regularization no longer follows the Bayesian framework, we point out that the lack-of-fit of the over-regularized estimate can increase

the deviance several orders of magnitude for every fixed number of clusters K, compared with the difference in deviance between different values of K! Thus, an aggressively regularized covariance estimate favors a small number of clusters K.

2.4.2 Starting values

Mixture model fitting implemented via the EM algorithm is sensitive with respect to starting values, and $\mathcal{MIX}_{\mathcal{L}}$ is no exception. We initialize the single-level fit, with M clusters, using the k-means clustering algorithm. Each single-level fit is initialized from several k-means clustering outcomes, and the best fit is reported.

As mentioned above, we explored various multi-level initialization schemes (e.g. forward search, where a 1st level cluster is split in $Dim(L)$, and backward search, where a cluster is joined to form a 1st level cluster $Dim(K)$). The best results were obtained with a backward search strategy. We initialize the multi-level fit with a total of M clusters. We run the EM algorithm with $K = M$ and $L_k = 1, \forall k = \{1, \cdots, M\}$. We then cluster the M cluster means and covariances into K clusters, using only parameters defined at the 1st level data dimension $Dim(K)$. This identifies clusters that can potentially form 1st level clusters, with sub-clusters defined over $Dim(L)$. The k-means clustering of the mean and covariance parameters from the M single-level fit identifies sub-cluster constellations $\mathbf{L}_K = \{L_k, k = 1, \cdots, K\}\}$, where $\sum_k L_k = M$. We run the multi-level EM algorithm from this initialization. To avoid convergence to local optima, we form at least B unique groupings of the M clusters into K 1st level clusters, and run the multi-level fit from all B initializations. The unique groupings are obtained by running k-means on the $Dim(K)$ parameter set repeatedly, and through random perturbations of the cluster allocations. It is absolutely necessary to run the multi-level clustering from several single-level initializations, and several groupings into K 1st level clusters, since the best single-level fit is not guaranteed to generate the best multi-level fit. In practice, we found that $B = 10$ alternative starting values for the single-level fit, and groupings into multi-level initializations, were sufficient. Since the above initialization procedure starts running the multi-level fit with starting values

obtained from an unconstrained fit, the first iterations of the profile EM (for $L_k > 1$ for any k, or after subset selection) in general decreases the likelihood. After $1-5$ iterations, the EM steps reverse direction, and converge toward a constrained solution. In general, the multi-level fit converged after fewer than 50 iterations, whereas the EM run after subset selection converged after 25 iterations or less.

3 Application to Data

3.1 The proliferating cell-line data

We apply the $\mathcal{MIX}_{\mathcal{L}}$ model with subset selection to a data set of proliferating stem cell lines (Goff et al. (2006)). At the onset of the culture study, the two cell-lines are virtually identical, and are believed to exist in a state of "pre-conditioning" or "pre-programming". Thus, sets of neuron-specific and glia-specific genes are active, and will determine the cell-fate of the clones. In this experiment, the differentiation process of neuron and glia has been accelerated via the withdrawal of a growth factor (FGF). Each culture was stained for neuron and glia specific markers. Each of three cultures of each type was followed over 3 days. mRNA was extracted for array analysis at $t = 0, 1$ and 3 days after the withdrawal of the growth factor.

The ABI system rat-chips, with 28,000 probes, were used for the array experiments. Of these probes, we studied a subset of 15,111 probes with complete annotation (a well known starting point of the coding region, and promoter). Preliminary significance analysis of the expression data identified 780 genes of the 15,111 as being significantly differentially expressed between the cell lines and/or time points at FDR 1% (using the Welch F-test and the Benjamini-Hochberg p-value corrections). For each of the 780 selected genes, we computed the mean gene profile across replicates, and standardized the mean profiles to have standard deviation 1, with a baseline of expression 0 for $t = 0$ in the glial like population. The final data set to be analyzed is thus of dimension 780 by 5. We denote gene expression in the glial-like population (L2.3) by $\mathbf{x}$, where $Dim(x)$

is 2 (for $t = 1$ and $t = 3$). We denote the gene expression in the neuron-like population by $\mathbf{y}$, where $Dim(y)$ is 3 ($t = 0, 1$ and $t = 3$).

3.2 Subset selection of cluster model profiles

We begin by exploring the impact of subset selection and specific parameterizations on clustering. As stated in the introduction, among the several scientific questions of interest were given to us by the biologists (Goff et al. (2006)) were; (a) How do the time course profiles of the glial-like (L2.3) and neuron-like (L2.2) cell-lines differ?; (b) Are there sets of genes for which the expression converges(diverges) between the glial-like and neuron-like cell populations?; (c) How dominant is the "pre-programming" effect? We introduced parameterizations W_I, W_{II} and W_{III} (section 2) to address these questions.

K	$W_I: \sum_k 1\{\beta_k = 0\}$	$W_{II}: \sum_k 1\{\beta_k = 0\}$	$W_{III}: \sum_k 1\{\beta_k = 0\}$
5	2	5	7
6	6	4	7
7	4	4	7
8	2	5	5
9	3	6	6
10	5	7	9
11	5	7	7
12	6	8	10

Table 1: Number of coefficients set to 0 by subset selection for the single-level fits with the three parameterizations.

In Figure 2 (c) we depict the BIC curves obtained for various numbers of clusters K in a single-level fit. The solid line is the BIC curve obtained without subset selection (i.e. a standard gaussian mixture model). The dashed and dotted lines are annotated with "1", "2" and "3", referring to the three parameterizations W_I, W_{II} and W_{III} respectively. Across all numbers of clusters, the W_{III} (cell fate pre-programming) parametrization is the most efficient, as indicated by the lower BIC values. The sparsity of each model is summarized in Table 1. With an efficient parametrization, both $K = 8$ and $K = 9$ are almost equally competitive. The W_{III} parametrization identifies cluster

Single-level		Multi-level		
K	$\sum_k 1\{\beta_k = 0\}$	(M, K)	$\sum_{kl} 1\{\beta_{kl} = 0\}$	multi-level constraints
5	7	(5,4)	5	2
6	7	(6,5)	11	2
7	7	(7,6)	2	2
8	5	(8,7)	4	2
9	6	(9,7)	4	4
10	9	(10,8)	4	4
11	7	(11,8)	11	6
12	10	(12,10)	6	4

Table 2: Number of coefficients set to 0 by subset selection for the single- and multi-level fits using parametrization W_{III}.

profiles that are static between $t = 0$ and $t = 1$, indicating a later developmental activity in one or both cell lines (e.g. clusters 2, 6) (see Figure 2 (a)).

Ultimately, the choice of parametrization should be guided by the scientific questions. Here, we had multiple questions to consider, and focus on the most efficient parametrization in our discussion. A more thorough comparison of the different parameterizations in a biological context is beyond the scope of this paper, but will be the focus of our future collaborative research with Professor R. Hart, Department of Neuroscience and Molecular Cell Biology at Rutgers University.

3.3 Multi-level model-based clustering of the cell-line study.

In Figure 2 (a) we depict the clustering outcome of a single-level fit using parametrization W_{III} (centered on cell fate pre-programming). As can be seen from the figure, the glial like population exhibits larger time differential effects than the neuron like population. Furthermore, for some clusters (e.g. 3 and 4), the glial like cluster expression profiles almost coincide, whereas the neuron cluster profiles differ substantially. To identify neuron specific variations, we will thus treat the glial like population data as the 1st level in the $\mathcal{MIX}_{\mathcal{L}}$ model.

In Figure 2 (d) we show the additional efficiency of parametrization from both

within-cluster and *between*-cluster comparisons. With the exception of the case with 5 total clusters, the multi-level fit always produces a lower BIC value. With these two levels of subset selection (within and between clusters), a total of $M = 9$ is in fact selected. That is, by using an efficient parametrization we gain one more cluster. One can view this as a re-allocation of model complexity. In model selection we aim to balance the fit and model complexity (number of parameters). By setting some cluster parameters to 0 (within-cluster subset selection), and letting some cluster share parameters at the 1st level (between-cluster parameter constraints), we save on complexity and can "afford" to form another cluster. In Table 2 we summarize the results on model selection, listing for each cluster the number of parameters set to 0 by a within-cluster profile subset selection, as well as the number of parameter constraints by the multi-level fit. For this data set, the larger gains are made when the number of clusters increase. For example, 11 out of 49 parameters were set to 0 (or constrained) in the $(M = 11, K = 8)$ multi-level fit.

The cluster profile that is the most unique in the multi-level fit is cluster 9 (Figure 2 (b) compared with (a)), which as we shall see in the discussion below provides some interesting insight into neuron specific activity. In addition, we have identified two groups of gene clusters (3 and 4, 5 and 6) for which the glial like population exhibits identical expression patterns, and a sub-division of genes exhibit radically different expression patterns in neurons.

3.4 Interpreting the clustering outcome

3.4.1 Examining the gene functional annotation of identified clusters

The efficient model description generated by the $\mathcal{MIX}_{\mathcal{L}}$ model gained us one extra cluster compared with the single-level fit. In addition, each cluster profile was described using a sparse representation if the data supported it. Clusters 3 and 4, as well and clusters 5 and 6, formed sub-clusters for which the expression pattern coincided in the glial like population, but differed substantially in the neuron like population. In Tables

4-5 the top 10 significant GO categories are reported for each of the 9 clusters. The GO terms were identified using GOstat (Beissbarth and Speed (2004)).

Among all clusters (780 genes), developmental terms and neurogenesis are over-represented compared with all annotated probes (15.111) on the array (Table 4 top).

Clusters 1 and 2: Cluster 1 corresponds to a set of genes that start out at baseline for both cell-lines, i.e. there is no pre-programming activity. In the glial like population, the expression of these genes increase rapidly over the course of the experiment. In table 4 (middle) we see that these genes are in fact annotated as specific to gliogenesis. The set of genes in cluster 2 are always overexpressed in the glia population compared with neurons, and the expression in glia increases over time. Table 4 (bottom) identifies this set of genes related to astrocyte formation (one type of glia), as well as transporter activity (of which chloride transport is a glial function).

Clusters 3 and 4: These clusters form a set of sub-clusters with neuron specific differential expression. To interpret these clusters, we rely on the following fact: it is known (from staining experiments) that the glial like cultures are heterogeneous. That is, in the cultures labeled "glial like" we see a mixture of glia and neurons. In contrast, the neuron population is largely homogeneous, and almost all cells in these cultures become neurons.

Cluster 3 represents genes that start off high in neurons, whereas the set of genes in glia population approach (from below) neuron specific levels of activity. Cluster 3 thus highlights genes that are believed to be specific to neuron formation. These genes are activated in the glia culture among cells that converge to neurons (Goff et al. (2006)). Looking in Table 5, GO categories that are overrepresented in cluster 3 correspond to neuron and neurite development, as well as activation of other neuron maturation processes (e.g. regulated by NFkappa-B). The neuron population has been 'pre-programmed' to this cell fate, and these genes are thus highly expressed throughout the experiment for these cultures.

Cluster 4 represent genes that start off more highly expressed in the neuron population. In the glial like population we again pick up the gene activity associated with the

sub-population converging to neurons. For these genes, activity is increasing in both populations. The GO categories associated with this cluster (Table 5) include growth cone, cytoskeleton, and microtubule binding, which are associated with dendrite formation (Charych et al. (2006)). Dendrites are part of the more complex neuron structure which explains the later activity of these genes compared with the more basic neuronal developmental processes identified in cluster 3.

Clusters 5 and 6: Clusters 5 and 6 again correspond to sub-clusters that are specific to activity in the neuron population. Cluster 5 corresponds to an overall higher activity in neurons compared with glia, and this activity is decreased in both populations. Cluster 6 corresponds to genes whose activity is always lower in neurons compared with glia, where again the glial activity is decreasing. In cluster 6, the glial gene expression is converging toward the neuron expression, suggesting that these genes are (de-)activated in the sub-population of cells in the glial population that form neurons. Cluster 5 is associated with acid metabolism, whereas cluster 6 is associated with acid synthesis. Acid metabolism is a process by which neurons generate neurotransmitters. Glial cells are believed to synthesize some acids that assist in neuron development and migration. Therefore, one can largely associate genes in cluster 5 with neuron specific activity, which explains the under expression in glia.

Cluster 7, 8 and 9: Cluster 7 corresponds to a more rapid increase in expression in the neuron population compared with glia (as indicated by the selected cluster model with no time effect in neurons between $t = 1$ and $t = 3$). This cluster is the most sparsely populated, with a large cluster variance. The GO terms associated with these clusters are not easy to interpret, with the exception of "morphogenensis". Cluster 8 is associated with expression upregulated in the neuron population compared with the glial population at the onset. The glial expression is slowly converging toward the neuron population. The top GO categories associated with cluster 8 are primarily centered on high level neuron functions (e.g. synaptic transmission). Cluster 9 consists of genes that are upregulated in neurons compared with glia at all times. The top GO categories in this cluster are linked to phosphorus binding. Phosphor is an activator of

BDNF binding, a primary regulator of dendritic branching at the cell body (primary branching). If we compare clusters 9 and 4, we see that primary branching (cluster 9) is activated early in neurons ($t = 0$) and then decreasing, whereas genes associated with dendritic formation and higher levels of branching (cluster 4) is associated with increasing gene expression over the course of the experiment.

3.4.2 Mining the clustering results

Regulation of gene expression in a condition specific manner heavily relies on the activities of the transcription factors, i.e., DNA binding proteins, and mainly on their recognition of DNA in a sequence specific manner. The sites that the transcription factors bind to on DNA are usually 5-20 base pairs long and are referred to as DNA binding motifs or regulatory motifs. Identification of these sites is a challenging and not completely solved computational biology problem. Recently, several methods (Bussemaker et al.; 2001; Keleş et al.; 2002; Conlon et al.; 2003) illustrated that addressing this problem in a feature/variable selection framework is a powerful way of elucidating experiment/class specific binding sites. In these approaches, the key idea is to use regulatory motifs as covariates and generally gene expression (expressed versus not expressed) as an outcome of interest. Then, a linear regression model is typically built to link the motifs to the outcome. More recently, non-parametric regression approaches like logic regression (Ruczinski et al.; 2003) and MARS (Friedman; 1991) are also employed (Keleş et al.; 2004; Das et al.; 2004) instead of linear regression models.

In our analysis, we use the cluster assignment of each gene as a class label and consider all pairwise comparisons of the clusters in a logistic regression framework. Covariates in these regression models are based on the transcription factor database TRANSFAC (Wingender; 1994). For each gene, we construct a set of covariates utilizing the position specific probability matrix (PSPM) representations of the regulatory motifs. This representation corresponds to a 4 by length of the motif matrix where each (i,j)th entry corresponds to the probability of observing the ith nucleotide at the jth position of the motif (see Stormo (2000) for a comprehensive review of binding site

representations). In order to construct the covariates, we extract first 1000 base pairs upstream of the transcription start site, i.e., regulatory region, for each gene. Then, these regions are scanned by each of the 795 regulatory motif PSPMs from TRANSFAC using the PATSER tool (Hertz and Stormo; 1999). As a result, we obtain, for each subsequence in the upstream sequence, a likelihood ratio score representing the likelihood of the subsequence under the regulatory motif model as opposed to a background model that assigns (0.3, 0.2, 0.2, 0.3) probabilities to the nucleotides A, C, G, and T, respectively.

The score of the best matching subsequence within the regulatory region is used as a covariate. Due to the high dimensional covariate space, elaborate variable selection schemes are required to identify the most relevant features.We utilize the recently developed `GLMpath` algorithm of (Park and Hastie; 2006). `GLMpath` fits L_1 regularized generalized linear models by solving the following minimization problem:

$$\hat{\beta}(\lambda) = \text{argmin}\{-\log L(\mathbf{y};\beta) + \lambda||\beta||_1\},$$

where λ is the regularization path and $L(\mathbf{y};\beta)$ represent the logistic regression likelihood parameterized by regression coefficients β in our framework. In our application, the regularization parameter is based on 5-fold cross-validation.

The number of discriminating position weight matrices identified for each pairwise comparison ranged from 0 to 9. Since TRANSFAC does not span the space of all position weight matrices relevant for rat, we indeed expect some of the pairwise comparisons not to have any discriminating position weight matrices. It has been previously noticed that although a linear regression analysis of gene expression as a function of regulatory sequences can elucidate major regulatory sequences affecting gene expression, such an analysis has typically low predictive power (Bussemaker et al.; 2001; Keleş et al.; 2002). Using a summary measure of gene expression, namely the clustering results, behaves similarly. Although we consider all pairwise comparisons, our main interest lies in the comparisons between the second level sub-clusters of the multi-level fit. As

depicted in Figure 2 (b), sub-clusters 3 and 4 and sub-clusters 5 and 6 are obtained via a split in the second cell line. Examining the position weight matrices selected for these comparisons, we note that M00133 matrix which is identified in the comparison of clusters 3 and 4 corresponds to transcription factor Tst-1. Tst-1 is a member of the POU domain gene family and is expressed in specific neurons and in myelinating glia in the mammalian nervous system. This transcription factor, also called MeF2, has been identified by our collaborators in an independent biochemistry experiment (Goff et al. (2006)). MeF2 is believed to be a target of a neurogenesis regulating microRNA, and its association with a neuron specific expression pattern in our study lends support to this biological hypothesis. Further study of the identified neuron-specific transcription factors are now underway in collaboration with Professor R. Hart at Rutgers.

4 Simulations studies

The clustering analysis of the proliferating cell-line data was applied to a set of significant genes only. In addition, prior to clustering, each gene expression profile was standardized with a fixed baseline at 0 for the glial-like cell-line at $t = 0$. The dimensions of the data set that was clustered was thus (780 by 5).

We simulate (780 by 5) multi-variate normal data, from several realistic scenarios. We use the estimated best single-level (SF) and multi-level (MF) fits (see section 3) to generate data on which we validate; (a) the selection of the number of clusters at each level; and (b) the specific subset model for each cluster (the non-zero coefficients). The best single level model is referred to as $Mod(1)$, and the best multi-level model as $Mod(2)$.

$Mod(1)$ is a single-level model with $K = 8$ clusters. The cluster means for this model are depicted in Figure 2 (a). The cluster means are parameterized with $5 * 8$ coefficients, and 5 parameters were eliminated (set to 0) by subset selection. $Mod(2)$ is the multi-level model with $M = 9$ clusters total, and $K = 7$ clusters at the 1st level. 4 parameters were eliminated (set to 0) by the subset selection, and 4 parameters

eliminated by the multi-level structure of the model. We simulate $Mod(1)$ and $Mod(2)$ data sets of the same dimensions as the original data set. We then perform single- and multi-level fits, as well as subset model selection on each of 50 simulated data sets. For each simulated data set, we record the selected number of clusters. We also compare the selected subset model (for the true number of clusters) to the true model, and record the total number of selection errors (the number of coefficients erroneously set to 0, or non-zero). We also compute the BIC of each fit, before and after subset selection.

In Figure 4 and Table 3 we summarize the results from the simulations. Figure 4 (a) shows that indeed the BIC is always reduced after model selection, even after the EM steps are rerun with the selected parameter constraints. Thus, performing subset selection on a cluster by cluster basis, using the local BIC, always produces a better model in terms of the BIC validation index. In Figure 4 (b) we depict a histogram of the total number of selection errors (across all clusters) for the 50 simulated data sets. In the case of the single level model ($Mod(1)$) (top panel), the multi-level fit (MF) generates fewer selection errors than the single-level fit. This is an intriguing result, given that the multi-level fit for which these errors are compared is constrained to only have $L_k = 1$, i.e. no sub-clusters. The reason for the improved selection performance is that we visit internal (1st level) clusters, and leaf (2nd level) clusters separately, and are thus performing subset selection on $2 * (K = 8)$ clusters in the multi-level fit, compared with $K = 8$ clusters in the single-level fit.

In Figure 4 (c) and (d) (lower panel), we depict the BIC reduction of the multi-level fit compared with the single-level fit, before and after the selection of the number of clusters, as well as after subset selection. In Figure 4 (c) we illustrate the results for the $Mod(1)$ (single-level fit is correct). We see that before subset selection, the single- and multi-level fits perform equally well (no difference in BIC value). After model selection, due to the increased number of clustered considered separately in the selection procedures (as stated above), the multi-level fit improves on the single-level fit. In Figure 4 (d), we illustrate the results from the $Mod(2)$ simulation (multi-level

fit is correct). Here, the multi-level fit improves on the single-level fit both before and after selection. Occasionally, the multi-level fit will perform worse than the single-level fit. This is a direct result of the limitations of the simulations study. The multi-level fits require a more careful exploration across multiple starting values. However, for ease of computation, the single- and multi-level fits were only run from one starting value, which favors the single-level fit. Still, with the exception of a few rare cases, the multi-level fit provides a better solution for $Mod(2)$ data. The histograms in Figure 4 (b) (bottom panel) shows that the total number of selection errors is yet again smaller for the multi-level fit (MF).

	SF(Mod 1)	MF(Mod 1)		
	K=M	K=7	K=8	K=9
M=8	44	1	**43**	
M=9	6	0	2	**4**

	SF(Mod 2)	MF(Mod 2)				
	K=M	K=6	K=7	K=8	K=9	K=10
M=7	1	**1**	0			
M=8	7	**2**	**3**	1		
M=9	15	**0**	**7**	**8**	2	
M=10	27	**0**	**2**	**7**	**11**	6

Table 3: Top panel: The selected number of clusters with the single-level and multi-level fits for the $Mod(1)$ data. The correct $K = 8$. Both fitting strategies perform well, and the multi-level fit correctly identifies a single-level fit (bold face in table) in almost all cases. Lower panel: The selected number of clusters with the single- and multi-level fits for the $Mod(2)$ data. The correct $M = 9$, with 7 1st level clusters ($K = 7$). In almost all cases, the multi-level fit correctly identifies a sub-cluster structure (bold face in table), rather than single-level model.

In Table 3 we present the selected number of cluster for the $Mod(1)$ and $Mod(2)$ data sets, using the single-level and multi-level fits. In the case of $Mod(1)$ data, the multi-level fit in almost all cases identifies the single-level fit as the correct model structure. In the case of $Mod(2)$ data, the multi-level fit in almost all cases identifies a multi-level fit (with sub-clusters) as the correct model structure. In the case of $Mod(2)$,

both fitting strategies have trouble identifying the correct total number of clusters. The reason is the cluster 7 in $Mod(2)$ is sparsely populated. In some simulations, cluster 7 is split into 2 cluster, producing in a total of $M = 10$ clusters. Sometimes "genes" in cluster 7 are simply allocated to nearby clusters, producing a total of $M = 8$ clusters.

In summary, the multi-level fit can correctly identify a single-level model as well as a multi-level model. In addition, the BIC is much reduced if the multi-level structure of the data is accounted for. Subset selection also reduces the BIC, in both single-level and multi-level models. The multi-level model always produces more accurate selection results, in part because the subset selection is applied to 1st level and second level clusters separately. Our simulation illustrates the impact of an efficient representation of cluster profile models (both within and between clusters) on mixture model fitting.

5 Discussion

We have proposed a mixture model with multiple levels to more efficiently model multiple-factor experimental data. In addition, we proposed a subset selection method to generate sparse representations of cluster profiles, under various parameterizations. We illustrated on real and simulated data that these efficient representations of the clusters models can have a substantial impact on the fit of the data, significantly reducing the BIC of the optimal model fits. In addition, we showed that our multi-level mixture modeling approach with subset selection can correctly identify both single-level and multi-level data structures. In our simulation setting, the multi-level approach produced subset selection results closer to the correct subset model, in both the single- and multi-level setting.

Our multi-level approach identified interesting and biologically relevant groups of genes in the proliferating cell-line data. A more thorough study of our findings is now underway in collaboration with biologists at Rutgers university.

Efficient cluster model representations (multiple levels and subset selection) will have a larger impact in high-dimensional settings, e.g., time-course data with more

time points. It is in these cases that a multi-level approach with subset selection has the largest potential in substantially reducing the number of parameters in the model. In addition, while we did not consider efficient representations of the cluster covariances, this is another area in which modeling efficiency may be explored. Fraley and Raftery (2002) compared mixture models with parameterized cluster covariances. Incorporating covariance parametrization and subset selection into our multi-level approach is an interesting future research topic.

While we demonstrated our multi-level approach on a proliferating cell-line data, with a two-level factor of interest, the method can in theory be extended to more factors, and factors with more levels. However, the profile EM algorithm we proposed may not be as easily adapted to a more complex data structure. For now, we recommend a grouping of the factors of interest, creating a baseline factor level for the 1st level of the model, and modeling contrasts from the baseline at the second level. Our future research will focus on the development of multi-level models and estimation procedures with a modeling hierarchy beyond two levels.

Acknowledgement

We thank Professor Ron Hart, and members of the Hart lab for generously sharing their data with us, and for helping us with a preliminary interpretation of the analysis outcome.

RJ is partially supported by NSF grant DMS0306360. RJ is also supported by the USEPA-funded Environmental Bioinformatics and Computational Toxicology Center (ebCTC), under STAR Grant number GAD R 832721-010. This work has not been reviewed by and does not represent the opinions of the funding agencies.
SK is supported by a WARF grant from the University of Wisconsin, Madison.

List of Figures and Tables

Figure 1: (a) An illustration of a Mixture Model with 2 levels; Solid line (black and gray): Two 1st level clusters for cell-line 1. The two sets of dashed lines (black and gray) represent the corresponding sub-clusters (level 2) for cell-line 2. Thus, here $K = 2$, and $L_1 = 2$, $L_2 = 2$. (b) The "Dynamic DE (differential expression) parametrization". The α parameters model the time course expression profile for cell-line 1, whereas the γ parameters model the time course of *cell-line differential* expression.

Figure 2: (a) Cluster mean profiles of the best single-level fit ($K = 8$). The glial like population is depicted in the left panel, the neuron like in the right panel (parametrization W_{III}) (b) Cluster mean profiles of the best multi-level fit $K = 7$, $M = 9$, with two sets of sub-clusters (parametrization W_{III}). (c) The BIC curves obtained using the single-level fit. Solid line: no subset selection. Dashed and dotted curves annotated with the respective parametrization (W_I, W_{II}, W_{III}). The W_{III}-BIC curve is the lowest, indicating that the W_{III} parametrization is the most efficient for this data set. (d) The BIC curves obtained from the single- and multi-level fits, using the W_{III} parametrization. The multi-level fit always gives a lower BIC for the same total number of clusters (M). The numbers in the figures (M, K) refers to the total number of clusters, and the number of 1st level clusters respectively. The best BIC values is obtained with $M = 9$ clusters total, and $K = 7$ 1st level clusters.

Figure 3: The 9 clusters generated by the best $\mathcal{MIX}_{\mathcal{L}}$ fit.

Figure 4: (a) The BIC value of the full model minus the BIC value after subset selection for both simulation settings: $(Mod(1), Mod(2))$, and both fitting strategies (single- (SF) and multi-level (MF) fits). The BIC is always smaller after subset selection. (b) Histograms of the total number of subset selection errors for the $Mod(1)$

data (40 parameters total) (top panel) and the $Mod(2)$ data (41 parameters total) (lower panel). The multi-level fit produce fewer selection errors in both cases. (c) The BIC of the single-level fit minus the BIC of the multi-level fit for $Mod(1)$ data, before and after subset selection. After subset selection, the multi-level fit improves on the single-level fit, even when the single-level model is correct. (d) The BIC of the single-level fit minus the BIC of the multi-level fit for $Mod(2)$ data, before and after subset selection. The multi-level fit improves on the single-level fit in almost all cases.

Table 1: Number of coefficients set to 0 by subset selection for the single-level fits with the three parameterizations.

Table 2: Number of coefficients set to 0 by subset selection for the single- and multi-level fits using parametrization W_{III}.

Table 3: Top panel: The selected number of clusters with the single-level and multi-level fits for the $Mod(1)$ data. The correct $K = 8$. Both fitting strategies perform well, and the multi-level fit correctly identifies a single-level fit (bold face in table) in almost all cases. Lower panel: The selected number of clusters with the single- and multi-level fits for the $Mod(2)$ data. The correct $M = 9$, with 7 1st level clusters ($K = 7$). In almost all cases, the multi-level fit correctly identifies a sub-cluster structure (bold face in table), rather than single-level model.

Table 4: Top 10 GO categories of all clusters, and clusters 1 and 2.

Table 5: Top 10 GO categories for clusters 3 and 4.

Table 6: Top 10 GO categories for clusters 5 and 6.

Table 7: Top 10 GO categories for clusters 7, 8 and 9.

References

Banfield, J. D. and Raftery, A. E. (1993). Model-based gaussian and non-gaussian clustering, *Biometrics* **48**: 803–821.

Beissbarth, T. and Speed, T. P. (2004). Gostat: Find statistically overrepresented gene ontologies within a group of genes, *Bioinformatics* **20(9)**: 1464–1465.

Bussemaker, H., Li, H. and Siggia, E. (2001). Regulatory element detection using correlation with expression, *Nature Genetics* **27**: 167–171.

Celeux, G. and Govaert, G. (1993). Comparison of the mixture and the classification maximum likelihood in cluster analysis, *Journal of Statistical Computation & Simulation* **47**: 127–146.

Charych, E. I., Akum, B. F., Goldberg, J., Jornsten, R. J., Rongo, C., Zheng, J. Q. and Firestein, B. L. (2006). Activity-independent regulation of dendrite patterning by postsynaptic density protein psd-95, *Journal of Neuroscience* **26(40)**: 10164–76.

Conlon, E., Liu, X., Lieb, J. and Liu, J. (2003). Integrating regulatory motif discovery and genome-wide expression analy sis, *Proceedings of the National Academy of Sciences USA* **100**: 3339–3344.

Das, D., Banerjee, N. and Zhang, M. Q. (2004). Interacting models of cooperative gene regulation, *Proceedings of National Academy of Science, USA* **101**(46): 16234–16239.

Dempster, A. P., Laird, N. and Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algorithm, *JRSSB* **39**: 1–38.

Fraley, C. and Raftery, A. E. (2002). Model-based clustering, discriminant analysis, and density estimation, *Journal of the American Statistical Association* **97**: 611–631.

Fraley, C. and Raftery, A. E. (2004). Bayesian regularization for normal mixture estimation and model-based clustering, *Technical Report 486*, University of Washington.

Friedman, J. H. (1991). Multivariate adaptive regression splines, *Annals of Statistics* **19**: 1–141.

Friedman, J. and Meulman, J. (2002). Clustering objects on subsets of attributes, *Technical report*, Department of Statistics, Stanford.

Goff, L. A., Davila1, J., Jörnsten, R., Keles, S., Li, H., Grumet, M. and Hart, R. P. (2006). Co-regulation of a single mir-9 locus and the adjacent mef2c gene during neuronal differentiation in neural stem cells., *submitted to Journal of Neuroscience* .

Hastie, T. and Tibshirani, R. (1996). Discriminant analysis by gaussian mixtures, *Journal of the Royal Statistical Society - Series B* **58**(158-176).

Hertz, G. Z. and Stormo, G. D. (1999). Identifying dna and protein patterns with statistically significant alignments of multiple sequences, *Bioinformatics* **15**(7): 563–577.

Hoff, P. (2006). Model-based subspace clustering, To appear in Bayesian Analysis.

Jornsten, R. (2004). Clustering and classification based on the l1 data depth, *Journal of Multivariate Analysis* **90**: 67–89.

Jornsten, R., Vardi, Y. and Zhang, C.-H. (2002). *Statistical data analysis based on the L1norm and related methods*, Statistics for industry and technology, Birkhauser, chapter A Robust Clustering Method and Visualization Tool Based on Data Depth.

Kaufman, L. and Rousseeuw, P. J. (1990). *Finding Groups in Data: An introduction to cluster analysis*, Wiley, New York.

Keleş, S., van der Laan, M. and Eisen, M. (2002). Identification of regulatory elements using a feature selection method, *Bioinformatics* **18**(9): 1167–1175.

Keleş, S., van der Laan, M. and Vulpe, C. (2004). Regulatory motif finding by logic regression, *Bioinformatics* **20**(16): 2799–2811.

Law, M. H., Figueiredo, M. A. and Jain, A. K. (2004). Simultaneous feature selection and clustering using mixture models, *IEEE Pattern Analysis and Machine Intelligence* **26**(9): 1154–1166.

Li, J. (2005). Clustering based on a multi-layer mixture model, *Journal of Computational and Graphical Statistics* **14**(3): 547–568.

McLachlan, G. and Peel., D. (2000). *Finite Mixture Models*, 1 edn, Wiley, New York.

Park, M. and Hastie, T. (2006). An l1 regularization-path algorithm for generalized linear models. a generalization of the lars algorithm for glms and the cox proportional hazard model. `http://www-stat.stanford.edu/~hastie/Papers/glmpath.pdf`.

Raftery, A. and Dean, N. (2006). Variable selection for model-based clustering, To appear in the *Journal of the American Statistical Association*.

Ruczinski, I., C., K. and M.L., L. (2003). Logic regression, *Journal of Computational and Graphical Statistics* **12**(3): 475–511.

Stormo, G. D. (2000). DNA binding sites: representation and discovery, *Bioinformatics* **16**(1): 16–23.

Tadesse, M. G., Sha, N. and Vannucci, M. (2005). Bayesian variable selection in clustering high-dimensional data., *Journal of the American Statistical Association* **100**: 602–617.

Wingender, E. (1994). Recognition of regulatory regions in genomic sequences, *Journal of Biotechnology* **35**: 273–280. `http://transfac.gbf.de/`.

Yuan, M. and C.Kendziorski (2006). A unified approach for simultaneous gene clustering and differential expression identifcation. To appear in *Biometrics*.

Zhu, H. and Zhang, H. (2004). Hypothesis testing in mixture regression models, *Journal of the Royal Statistical Society - Series B* **66**(1): 3–16.

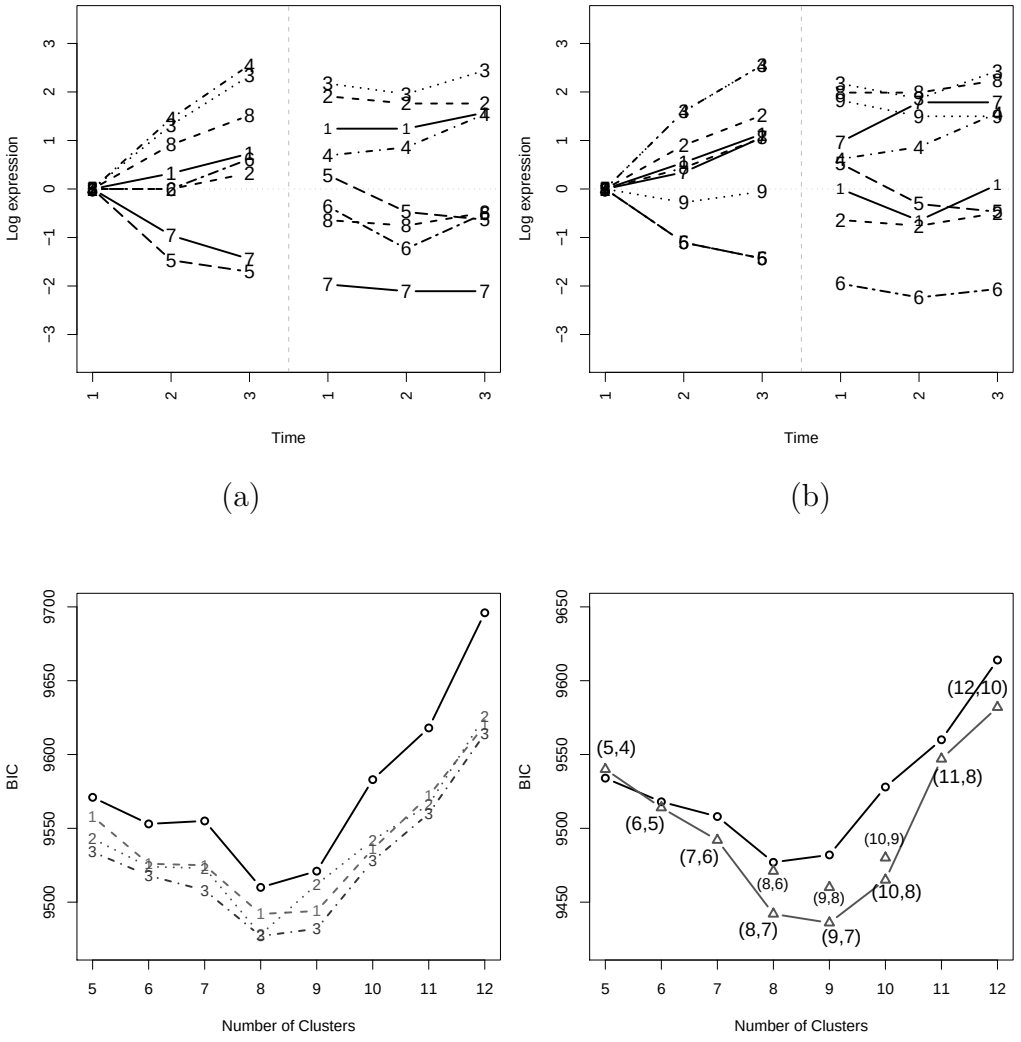

Figure 2: (a) Cluster mean profiles of the best single-level fit ($K = 8$). The glial like population is depicted in the left panel, the neuron like in the right panel (parametrization W_{III}) (b) Cluster mean profiles of the best multi-level fit $K = 7$, $M = 9$, with two sets of sub-clusters (parametrization W_{III}). (c) The BIC curves obtained using the single-level fit. Solid line: no subset selection. Dashed and dotted curves annotated with the respective parametrization (W_I, W_{II}, W_{III}). The W_{III}-BIC curve is the lowest, indicating that the W_{III} parametrization is the most efficient for this data set. (d) The BIC curves obtained from the single- and multi-level fits, using the W_{III} parametrization. The multi-level fit always gives a lower BIC for the same total number of clusters (M). The numbers in the figures (M, K) refers to the total number of clusters, and the number of 1st level clusters respectively. The best BIC values is obtained with $M = 9$ clusters total, and $K = 7$ 1st level clusters.

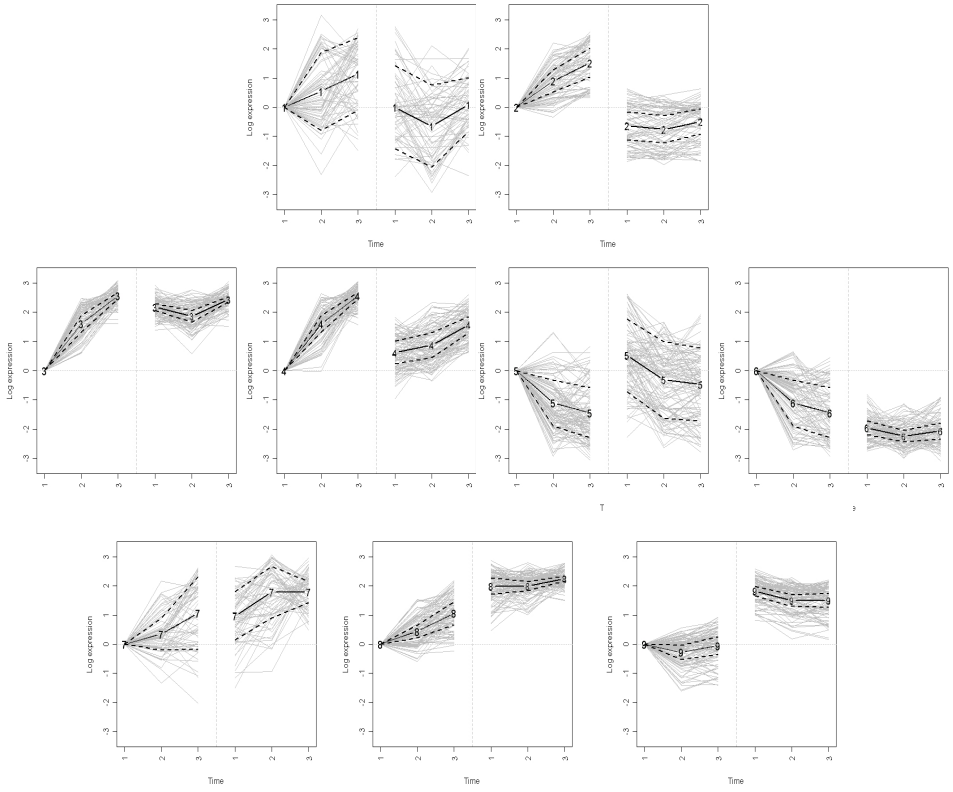

Figure 3: The 9 clusters generated by the best $\mathcal{MIX}_{\mathcal{L}}$ fit.

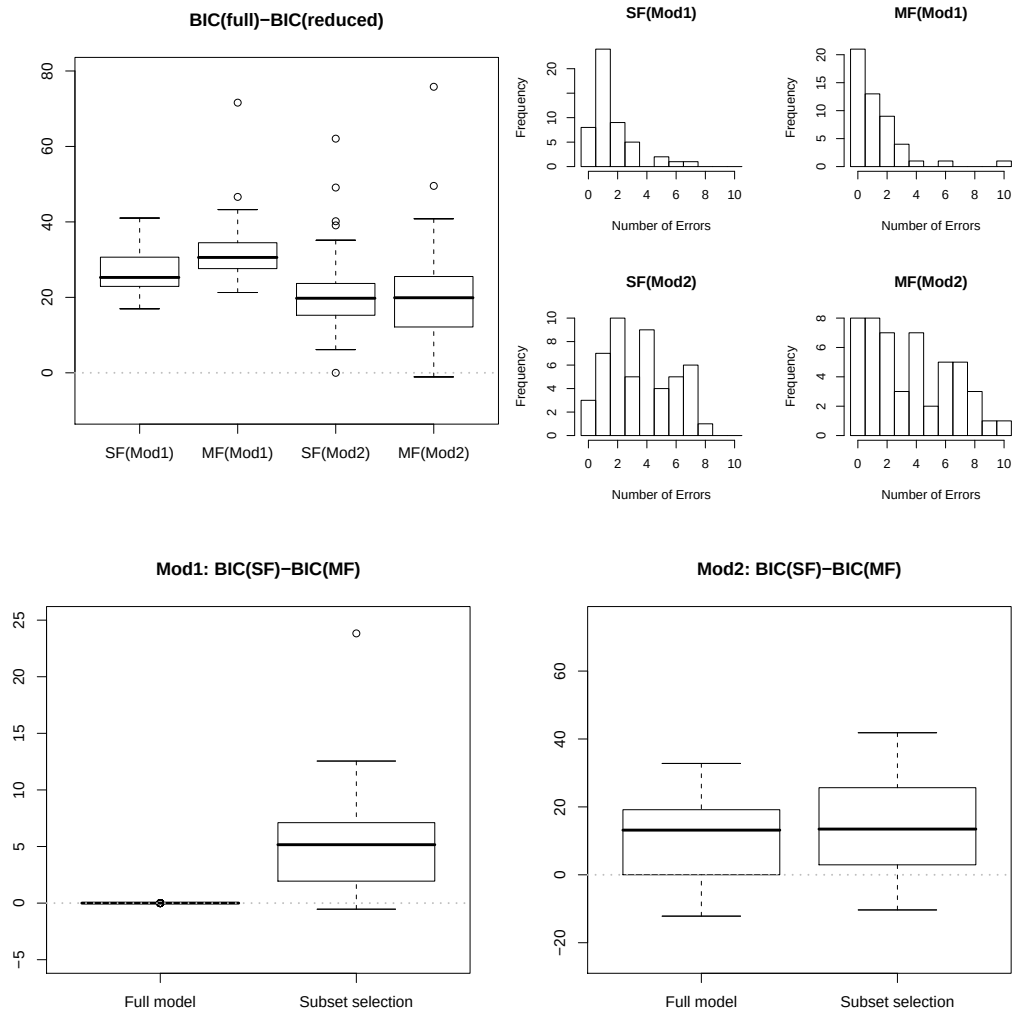

Figure 4: (a) The BIC value of the full model minus the BIC value after subset selection for both simulation settings: $(Mod(1), Mod(2))$, and both fitting strategies (single- (SF) and multi-level (MF) fits). The BIC is always smaller after subset selection. (b) Histograms of the total number of subset selection errors for the $Mod(1)$ data (40 parameters total) (top panel) and the $Mod(2)$ data (41 parameters total) (lower panel). The multi-level fit produce fewer selection errors in both cases. (c) The BIC of the single-level fit minus the BIC of the multi-level fit for $Mod(1)$ data, before and after subset selection. After subset selection, the multi-level fit improves on the single-level fit, even when the single-level model is correct. (d) The BIC of the single-level fit minus the BIC of the multi-level fit for $Mod(2)$ data, before and after subset selection. The multi-level fit improves on the single-level fit in almost all cases.

All clusters vs GO data base	
GO0048731	"System development"
GO0007399	"Nervous system development"
GO0030154	"Cell differentiation"
GO0006928	"Cell motility"
GO0051674	"Location of cell"
GO0040011	"Locomotion"
GO0022008	"Neurogenesis"
GO0051606	- "Detection of stimulus"
GO0009582	- "Detection of abiotic stimulus"
GO0030182	"Neuron differentiation"
Cluster 1 vs All clusters	
GO0006836	"Neurotransmitter transport"
GO0042063	"Gliogenesis"
GO0010001	"Glial cell differentiation"
GO0007399	"Nervous system development"
GO0031324	"Neg. regulation of cell metabolism"
GO0048737	"System development"
GO0006357	"Neg. reg. RNA polymerase transcription"
GO0001504	"Neurotransmitter uptake"
GO0048469	"Cell maturation"
GO0001764	"Neuron migration"
Cluster 2 vs All clusters	
GO0015290	"El.chem transport activity"
GO0015291	"Porter activity"
GO0015293	"Symporter actitivy"
GO0005416	"Amino acid symporter activity"
GO0048143	"Astrocyte formation"
GO0015103	"Anion transport activity"
GO0006820	"Anion transport"
GO0015380	"Anion exchange activity"
GO0015108	"Chloride transporter activity"
GO0015297	"Antiporter activity"

Table 4: Top 10 GO categories of all clusters, and clusters 1 and 2.

Cluster 3 vs All clusters	
GO0005694	"Chromosome"
GO0009966	"Reg. signal transduction"
GO0030900	"Forebrain development"
GO0007249	"NFkappa-B cascade"
GO0031175	"Neurite development"
GO0048666	"Neuron development"
GO0000785	"Chromatin"
GO0044427	"Chromosomal part"
GO0007242	"Intracell. signal cascade"
GO0007409	"Axonogenesis"
Cluster 4 vs All clusters	
GO0030427	"Site of polarized cone"
GO0030426	"Growth cone"
GO0015631	"Tubulin binding"
GO0005856	"Cytoskeleton"
GO0008017	"Microtubule binding"
GO0030018	"Z-disc"
GO0005886	"Plasma membrane"
GO0000267	"Cell fraction"
GO0044228	"Non-membrane-bound organelle"
GO0017111	"Nucleoside-triophasphate act."

Table 5: Top 10 GO categories for clusters 3 and 4.

Cluster 5 vs All clusters	
GO0006767	"Vitamin metabolism"
GO0005739	"Mitochondria"
GO0019752	"Carb. acid metabolism"
GO0006082	"Organic acid metabolism"
GO0031975	"Envelope"
GO0031967	"Organelle envelope"
GO0044237	"Cell metabolism"
GO0043170	"Macromolecule metabolism"
GO0009058	"Biosynthesis"
GO0006865	"Amino acid transport"
Cluster 6 vs All clusters	
GO0044272	"Sulfur compound biosynthesis"
GO0008652	"Amino acid biosynthesis"
GO0000097	"Sulfur amino acid biosynthesis"
GO0006092	"Pathway of carbohydrate metabolism"
GO0050794	- "Neg. reg. cell process"
GO0008217	"Blood pressure regulation"
GO0008202	"Steroid metabolism"
GO0005624	"Membrane fraction"
GO0005515	- "Protein binding"
GO0000267	"Cell fraction"

Table 6: Top 10 GO categories for clusters 5 and 6.

Cluster 7 vs All clusters	
GO0048729	"Morphogenesis"
GO0050874	"Tissue development"
GO0009605	"Response to external stimulus"
GO0016042	"Lipid catabolism"
GO0050875	"Organ. phys. process"
GO0050896	"Response to stimulus"
GO0008081	"Phospholiric dieter hydrolase activity"
GO0042330	"Taxis"
GO0006935	"Chemotaxis"
GO0005543	"Phospholipid binding"
Cluster 8 vs All clusters	
GO0044421	"Extracell. region"
GO0043235	"Receptor complex"
GO0004720	"Protein-oxidase activity"
GO0007270	"Nerve-nerve synaptic transmission"
GO0044238	- "Primary metabolism"
GO0005615	"Extracellular space"
GO0009653	"Morphogenesis"
GO0007271	Synaptic transmission
GO0005102	Receptor binding
GO0000902	Cellular morphogenesis
Cluster 9 vs All clusters	
GO0006797	"Phosphorus metabolism"
GO0006796	"Phosphate metabolism"
GO0006350	"Transcription"
GO0045449	"Reg. of transcription"
GO0006351	"DNA-dependent transcription"
GO0019219	"Reg. of nucleic acid metabolism"
GO0006468	"Protein amino acid phosphorylation"
GO0006464	"Protein modification"
GO0043412	"Biopolymer modification"
GO0044237	"Cellular metabolism"

Table 7: Top 10 GO categories for clusters 7, 8 and 9.