The Dissertation Committee for Carlos Tadeu Pagani Zanini
certifies that this is the approved version of the following dissertation:

Dependent Mixtures and Random Partitions

Committee:

Peter Müller, Co-supervisor

Mingyuan Zhou, Co-supervisor

Sinead Williamson

Yuan Ji

Dependent Mixtures and Random Partitions

by

Carlos Tadeu Pagani Zanini

DISSERTATION

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT AUSTIN

August 2019

Dedicated to my family and friends.

Acknowledgments

I wish to thank the multitudes of people who helped me. Time would fail me to tell of ...

Dependent Mixtures and Random Partitions

Publication No. _____

Carlos Tadeu Pagani Zanini, Ph.D.
The University of Texas at Austin, 2019

Co-supervisors: Peter Müller
Mingyuan Zhou

This work develops new methodology for Bayesian dependent mixture models and dependent random partitions with applications to biomedical data. A mixture model implies a random distribution over partitions by randomly assigning individual observations to latent subpopulations that correspond to the distinct components of the mixture. Subpopulations are typically homogeneous, but heterogeneous accross groups. In the biomedical applications studied here, the mixture components capture different levels of gene/protein expression, distinct stages of cellular development or the response to exposition to distinct drugs. Multiple forms of dependence are considered in order to more accurately model biological features of the studied applications, including dependence over time, dependence by arrangement on a tree and by shared match with paired cell lines.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Objectives and Outline

This work develops new methodology for Bayesian dependent mixture models and dependent random partitions with applications to biomedical data. A mixture model implies a random distribution over partitions by randomly assigning individual observations to latent subpopulations that correspond to the distinct components of the mixture. Subpopulations are typically homogeneous, but heterogeneous accross groups. In the biomedical applications studied here, the mixture components capture different levels of gene/protein expression, distinct stages of cellular development or the response to exposition to distinct drugs. Multiple forms of dependence are considered in order to more accurately model biological features of the studied applications, including dependence over time, dependence by arrangement on a tree and by shared match with paired cell lines

Summary and contributions

1. In Chapter 2, we model changes in protein expression after cell lines are exposed to drugs (protein inhibitors) in an reverse phase protein array (RPPA) experiment. We allow for clusters of proteins with different

treatment effects, and allow these clusters to change over time. The proposed dependent random partitions define a refinement and coagulation of protein clusters over time. We implement the approach using a time-course RPPA dataset consisting of protein expression measurements under different drugs, dose levels, and cell lines.

Contributions: We developed a time-dependent random partition model that is defined by a sequence of random refinements and coagulations at random change points. The model includes monotonicity as implied by the application. In the motivating application, such dependence accounts for the identification of the proteins that are affected by drugs, although it can also be used in different applications that exhibit similar patterns.

2. In Chapter 3, we introduce dependent mixture models when the cluster locations are naturally connected by a spanning tree. The motivating application is inference for cell lineage data on the basis of single cell RNA sequencing (scRNAseq) data for cells across different levels of cell differentiation. The terms of the mixture model are interpreted as representing distinct cell types, including a known root cell population and final differentiated cells. We propose prior models based on prior shrinkage of a minimum spanning tree (MST) of cluster centers.

   Contributions: We develop a dependent mixture model where the dependence arrises from the nature of the components as the nodes of an underlying latent random tree. The dependence is represented by a regularization factor in the prior distribution of the locations of the

nodes and it penalizes over-complex tree structures.

3. In Chapter 4, we construct a novel Bayesian statistical approach for matching patient samples with cell lines. We propose a statistical approach that seamlessly combines the output of the Bayesian mixture model based on a proposal by Parmigiani et al. (2002) with a novel two-way Bayesian non-parametric (BNP) mixture model that is constructed as an extension of a BNP bi-clustering model of Lee et al. (2013).

   Contributions: The research described in that Chapter makes two important methodological contributions in Bayesian non-parametrics: (i) the seamless integration of the (modified) probability of expression (POE) model for noise reduction and the nested bi-clustering approach; (ii) we introduce a novel random structure to allow probabilistic modeling of co-clustering between proteins/genes and patients based on profile similarities via dependent priors on partition models.

Finally, in Chapter 5 we present conclusions and future work. Appendix A contains a list of well known probability distributions with the parameterization that is used throughout the thesis. Appendix B contains additional information that details the implementation of the MCMC algorithm that is discussed in Chapter 2, as well as details on the use of AIC and BIC to select the number of clusters. Appendix C describes details for the MCMC algorithm to sample from the posterior distribution under the two models proposed in Chapter 3: h-MST and s-MST. Finally, Appendix D contains the full con-

ditionals for the Metopolis within Gibbs algorithms that are used to sample from the posterior distribution under the models described in Chapter 4 POE model and uder the NobLoc model with matching of cell lines and patients.

1.2 The Bayesian Inference Framework

In this section, we introduce notation by way of a brief review of Bayesian inference for parameter estimation and prediction.

Consider a random variable Y with an assumed probability distribution that is indexed by a parameter vector $\boldsymbol{\theta}$. With the objective of understanding the probabilistic behavior of Y, a random sample $\boldsymbol{y} = y_1, \ldots, y_n$ is collected from Y from which we produce estimates of $\boldsymbol{\theta}$. This procedure works because the observed data carries information about the parameter $\boldsymbol{\theta}$ which is mathematically coded in the likelihood function $l(\cdot \ ; \ \boldsymbol{y}) : \boldsymbol{\Theta} \rightarrow \mathbb{R}^+$, defined as $l(\boldsymbol{\theta} \ ; \ \boldsymbol{y}) = p(\boldsymbol{y} \mid \boldsymbol{\theta})$, where $\boldsymbol{\Theta}$ is the parameter space and $p(\boldsymbol{y} \mid \boldsymbol{\theta})$ is the density function (or the probability mass function) of $\boldsymbol{y}$. The likelihood function can therefore be interpreted as a measurement of plausibility for $\boldsymbol{\theta}$ in the light of the observed data $\boldsymbol{y}$.

Under the Bayesian paradigm, subjective prior information about $\boldsymbol{\theta}$ is also considered. Such information is mathematically represented by the prior distribution $\pi(\boldsymbol{\theta})$ which is specified unconditionally on the observation of the data. Bayes theorem establishes the use of prior and likelihood to update uncertainty about $\boldsymbol{\theta}$.

Bayes theorem: Let $\boldsymbol{\theta} \in \boldsymbol{\Theta}$ be the parameter, $p(\boldsymbol{\theta})$ the density or probability mass function a priori, and $\boldsymbol{y}$ the vector of observations with likelihood $l(\boldsymbol{\theta}; \boldsymbol{y}) = p(\boldsymbol{y} \mid \boldsymbol{\theta})$. Then, the posterior distribution is given by

$$p(\boldsymbol{\theta} \mid \boldsymbol{y}) = \frac{p(\boldsymbol{y} \mid \boldsymbol{\theta})p(\boldsymbol{\theta})}{\int p(\boldsymbol{y} \mid \boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta}} \propto p(\boldsymbol{y} \mid \boldsymbol{\theta})p(\boldsymbol{\theta}),$$

where the product $p(\boldsymbol{y} \mid \boldsymbol{\theta})\pi(\boldsymbol{\theta})$, as well as any of its multiples by any factor that does not depend on $\boldsymbol{\theta}$, is known as the kernel of $\pi(\boldsymbol{\theta} \mid \boldsymbol{y})$.

All information on the parameter $\boldsymbol{\theta}$ after seeing the data is contained in the posterior distribution with associated density (or probability mass function) $p(\cdot \mid \boldsymbol{y}) : \boldsymbol{\Theta} \to \mathbb{R}^+$. The posterior distribution is used to calculate estimates of the parameters as well as to make predictions for new observations $\boldsymbol{y}^*$ through the predictive distribution

$$\begin{aligned} p(\boldsymbol{y}^* \mid \boldsymbol{y}) &= \int_{\Theta} p(\boldsymbol{y}^* \mid \boldsymbol{\theta}) \; dp(\boldsymbol{\theta} \mid \boldsymbol{y}) \\ &= \begin{cases} \int_{\Theta} p(\boldsymbol{y}^* \mid \boldsymbol{\theta})p(\boldsymbol{\theta} \mid \boldsymbol{y})d\boldsymbol{\theta}, & \text{(continuous case)} \\ \sum_{\boldsymbol{\theta} \in \Theta} p(\boldsymbol{y}^* \mid \boldsymbol{\theta})p(\boldsymbol{\theta} \mid \boldsymbol{y})d\boldsymbol{\theta}, & \text{(discrete case).} \end{cases} \end{aligned}$$

The predictive distribution can be interpreted as an average of values from $p(\boldsymbol{y}^* \mid \boldsymbol{\theta})$ weighted by the posterior $p(\boldsymbol{\theta} \mid \boldsymbol{y})$ on the observed data. The predictive distribution does not depend on $\boldsymbol{\theta}$ in its analytical form.

1.3 Bayesian Mixture Models

A large class of attractive models in Bayesian inference, especially in biomedical research problems, are hierarchical and related mixture models. Mixture models are probabilistic models obtained from the integration of a parameterized probability density (or probability mass function) with respect to a mixing measure on the parameter. For example, Gaussian mixture models are obtained as

$$p(\boldsymbol{y} \mid \boldsymbol{\theta}) = \int N(\boldsymbol{y} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma})\ dG_{\boldsymbol{\theta}}(\boldsymbol{\mu}, \Sigma), \tag{1.1}$$

where $N(\boldsymbol{x} \mid \boldsymbol{a}, \boldsymbol{B})$ denotes the density of a (multivariate) Gaussian distribution with mean $\boldsymbol{a}$ and covariance matrix $\boldsymbol{B}$ evaluated at $\boldsymbol{x}$. The mixing measure G is typically parameterized by unknown parameters $\boldsymbol{\theta}$, resulting in $p(\boldsymbol{y} \mid \boldsymbol{\theta})$ also being parameterized by $\boldsymbol{\theta}$. The model specification is completed by specifying a prior on $\boldsymbol{\theta}$.

Many different models $p(\boldsymbol{y} \mid \boldsymbol{\theta})$ can be written as in (1.1), depending on the choice of the mixing measure $G_{\boldsymbol{\theta}}$ and the prior on $\boldsymbol{\theta}$. We focus on cases where the integrand in (1.1) is Gaussian, although any other distribuitions could also be considered.

Example 1.3.1. *(Discrete Gaussian mixture model) In the case of a discrete mixing measure* $G_{\boldsymbol{\theta}}(\cdot) = \sum_k w_k I_{\mu_k, \Sigma_k}(\cdot)$ *with* $I_x(\cdot)$ *denoting a unit point mass (Dirac measure) at* x*, we have* $\boldsymbol{\theta} = (w_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, k = 1, 2, \ldots)$ *and the mixture reduces to* $p(\boldsymbol{y} \mid \boldsymbol{\theta}) = \sum_k w_k N(\boldsymbol{y}; \boldsymbol{\mu}_k, \Sigma_k)$.

Example 1.3.2. *(Student-t as a Gaussian scale mixture) Consider the unidimensional case $y \in \mathbb{R}$, with $p(y \mid \mu, \sigma^2) = N(y \mid \mu, \sigma^2)$. If $G_{\alpha,\beta}(\cdot)$ is the Gamma(α, β) distribution, then we get a location-scale Student-t$(2\alpha, \mu, \beta)$:*

$$
\begin{aligned}
p(y \mid \mu, \alpha, \beta) &= \int N(y \mid \mu, \sigma^2) dG_{\alpha,\beta}(\sigma^{-2}) \propto \\
&\propto \int_0^{+\infty} (\sigma^{-2})^{\frac{1}{2}} \exp\left\{-\frac{1}{2}(y-\mu)^2\sigma^{-2}\right\} \times (\sigma^{-2})^{\alpha-1} \exp(-\beta\sigma^{-2}) d(\sigma^{-2}) \propto \\
&\propto \left[\frac{\frac{(y-\mu)^2}{\beta} + 2\alpha}{2\alpha}\right]^{-\frac{2\alpha+1}{2\alpha}} .
\end{aligned}
$$

Example 1.3.3. *(Laplace as a Gaussian scale mixture) Consider the univariate case $p(y \mid \mu, \sigma^2)$ with mixing measure $G_\lambda(\sigma^2)$ being the Exp $\left(\frac{\lambda^2}{2}\right)$ distribution. Then $(y \mid \lambda, \mu) \sim$ Laplace$(\lambda, \mu, 1)$:*

$$
\begin{aligned}
p(y \mid \lambda, \mu) &= \int_0^{+\infty} N(y \mid \mu, \sigma^2) dG_\lambda(\sigma^2) \\
&\propto \int_0^{+\infty} (\sigma^2)^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}\sigma^{-2}(y-\mu)^2\right\} \exp\left\{-\frac{\lambda^2}{2}\sigma^2\right\} d\sigma^2 \\
&\propto \exp\left\{-\lambda|y-\mu|\right\} .
\end{aligned}
$$

An important application of the Laplace distribution as a scale mixture of normals arises in the Bayesian lasso (Park and Casella, 2008) variable selection approach where the Laplace prior is responsible for an L_1 regularization of the coefficients and the augmentation provided by the scale mixture representation

guarantees conjugacy for the full conditionals of the Gibbs sampler (section 1.4.2), therefore simplifying the algorithm.

We focus on discrete Gaussian mixtures as in Example 1.3.1. Implementing posterior simulation, the parameter space is augmented to include latent group assignment variables (or cluster membership indicators) δ_i, $i = 1, \ldots, n$ for observations y_i. The event $\{\delta_i = k\}$ indicates that observation i is sampled from the subpopulation k, i.e., $(\boldsymbol{y}_i \mid \delta_i = k, \boldsymbol{\mu}_k, \Sigma_k) \sim N(\boldsymbol{y}_i \mid \boldsymbol{\mu}_k, \Sigma_k)$. The probability vector $\boldsymbol{w} = (w_1, \ldots, w_K)$ then serves as prior for the cluster membership indicators: $P(\delta_i = k \mid \boldsymbol{w}) = w_k$.

The final step to define the Bayesian discrete Gaussian mixture model is to specify the prior for the atoms $(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)_{k=1}^K$ and for the probability vector $\boldsymbol{w}$. There are many possibilities for defining such priors. For finite discrete Gaussian mixtures ($K < \infty$) a common choice is a Dirichlet distribution for the weights: $\boldsymbol{w} \sim Dirichlet(\eta)$ and an i.i.d. conditionally conjugate prior for the atoms: $\boldsymbol{\mu}_k \sim N(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$, $\boldsymbol{\Sigma}_k \sim IW(\nu, \Psi)$. To summarize, the full Bayesian model in this case is:

$$(\boldsymbol{y}_i \mid \delta_i = k, \boldsymbol{\mu}_k, \Sigma_k) \sim N(\boldsymbol{y}_i \mid \boldsymbol{\mu}_k, \Sigma_k),$$
$$P(\delta_i = k \mid \boldsymbol{w}) = w_k,$$

and priors,

$$\boldsymbol{w} \sim Dirichlet(\boldsymbol{\eta}),\ \boldsymbol{\mu}_k \sim N(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0),\ \boldsymbol{\Sigma}_k \sim IW(\nu, \Psi). \tag{1.2}$$

Under the representation of the mixture model in equation (1.1) as an expectation with respect to a mixing measure G_θ it is natural to interpret (1.2) as a prior on the mixing measure G_θ. Prior probability models on random probability measures are also known aas non-parametric Bayes models (BNP) (Ferguson et al., 1992). In this sense, mixture models are naturally linked with BNP priors.

1.3.1 Bayesian non parametrics and mixture models

In contrast to parametric models, non-parametric models include infinitely many parameters which under the Bayesian framework requires a prior on a space of infinite dimensions. The main motivation for non-parametric models is the flexibility that is achieved in comparison with a parametric model with finite dimensional parameter space. In this section we will present some applications of Bayesian non-parametric (BNP) priors for mixture models.

We start with the arguably simplest non-parametric model on random probability measures: the Dirichlet process (DP) (Ferguson, 1973). If $G \sim DP(G_0, \alpha)$, we say that G is a random measure following a Dirichlet process with baseline probability measure G_0 on a set S and concentration parameter α. Ferguson (1973) defines $G \sim DP(G_0, \alpha)$ by defining probability assignments on partitions of S as $(G(B_1), \ldots, G(B_K)) \sim$ Dirichlet$((\alpha G_0(B_1), \ldots, \alpha G_0(B_K)))$ for any measurable partition $S = B_1, \cup \ldots \cup B_K)$ for any $K \in \mathbb{N}$. The author shows that the DP is well defined, meaning that there are no inconsistencies with the random assignment of probabilities through the DP.

However, the definition provided by Ferguson (1973) does not directly allow for an easy way of, for example, simulating such random measure. Simulation of a DP random measure is important to implement Bayesian inference in models involving DP's. Sethuraman (1994) provided a very simple and efficient way of sampling a DP. The procedure is called stick breaking representation and it works as follows. First generate a sequence of atoms $(\boldsymbol{\theta}_k)_{k=1}^{+\infty}$. For each $k \in \mathbb{N}$ sample $\beta_k \sim Beta(1, \alpha)$ and create a probability vector $\boldsymbol{w} = (w_k)_{k=1}^{\infty}$ as $w_1 = \beta_1$, $w_k = \beta_k \prod_{\ell<k}(1-\beta_\ell)$ for $K > 1$. This defines$G(\cdot) = \sum_{k=1}^{\infty} w_k I_{\boldsymbol{\theta}_k}(\cdot)$. The stick breaking construction by itself already gives valuable insights on $G \sim DP(G_0, \alpha)$ when G_0 is a continuous probability measure: (1) G is a discrete probability measure with infinite number of atoms; (2) the atoms are sampled i.i.d. from the baseline measure; (3) The atoms are a dense set in the support of G_0; (4) α controls the rate of decay of the weights w_k as $k \to \infty$.

1.4 Markov Chain Monte Carlo Posterior Simulation

In many important Bayesian models, it is not possible to carry out posterior integrals analytically. Alternatively , there are many numerical quadrature integration methods to approximate $p(\boldsymbol{y})$ such as trapezoids integration method, Simpson integration formula, Gauss Hermite quadrature and more (for a brief introduction, see for example Süli and Mayers 2003). Such methods usually work well when $\boldsymbol{\theta}$ is low dimensional because then the construction of the grid of points to integrate over can be reasonably distributed over the

parameter space $\boldsymbol{\Theta}$. However, in moderate dimension the construction of such a grid reaches a prohibitive computational cost.

In these cases, simulation based methods are used. One approach is Markov chain Monte Carlo (MCMC), which simulates a random Markov that is constructed to have the posterior $p(\boldsymbol{\theta} \mid \boldsymbol{y})$ as its invariant distribution. Then averages over the simulated states approximate posterior integrals as the algorithm iterates.

Next we describe two popular MCMC sampling schemes: the Gibbs sampler and the Metropolis-Hastings algorithm.

1.4.1 Metropolis Hastings

Consider a target probability distribution with density $\pi(\boldsymbol{x})$ and support $\boldsymbol{X} \subset \mathbb{R}^p$ from which we want to obtain a random sample by MCMC simulation. We suppose that $\pi(\boldsymbol{x})$ is analytically available up to a proportionality constant, i.e., $\pi(\boldsymbol{x}) = \pi^*(\boldsymbol{x})C^{-1}$ where C is unknown and the kernel $\pi^*(\boldsymbol{x})$ is available in analytic form with $\int_X \pi^*(\boldsymbol{x})d\boldsymbol{x} = C$. For example, $\pi(\boldsymbol{x})$ could be a posterior distribution in a Bayesian inference problem. We already saw that the kernel of the posterior distribution is analyticaly available when the prior $\pi(\boldsymbol{\theta})$ and the likelihood $p(\boldsymbol{y} \mid \boldsymbol{\theta})$ are analytically available.

The objective is to build an irreducible and aperiodic Markov chain with transition probability $p(\tilde{\boldsymbol{x}} \mid \boldsymbol{x})$ having invariant distribution $\pi(\boldsymbol{x})$. Such conditions guarantee the convergence of the Markov chain to its target invariant distribution $\pi(x)$. It is usually easy to build an irreducible aperiodic

Markov chain. A sufficient condition for invariance is the detailed balance condition.

Detailed balance condition: If $\pi(\tilde{\boldsymbol{x}})p(\boldsymbol{x} \mid \tilde{\boldsymbol{x}}) = \pi(\boldsymbol{x})\pi(\tilde{\boldsymbol{x}} \mid \boldsymbol{x}),\ \forall \boldsymbol{x},\ \tilde{\boldsymbol{x}}$ then $\pi(\boldsymbol{x})$ is the invariant distribution of the Markov chain with transition $p(\boldsymbol{x} \mid \tilde{\boldsymbol{x}})$. In this case, we say that $p(\tilde{\boldsymbol{x}} \mid \boldsymbol{x})$ satisfies the detailed balance condition with respect to the invariant distribution $\pi(\boldsymbol{x})$.

To create a transition $p(\tilde{\boldsymbol{x}} \mid \boldsymbol{x})$ that satisfies the detailed balance condition with respect to $\pi(\boldsymbol{x})$ we start with an initial proposal $q(\tilde{\boldsymbol{x}} \mid \boldsymbol{x})$ on $\mathcal{X}$ that is irreducible and aperiodic. The initial proposal will usually violate detailed balance condition, i.e., for some pairs $(\tilde{\boldsymbol{x}}, \boldsymbol{x}) \in X \times \mathcal{X}$, $\pi(\boldsymbol{x})q(\boldsymbol{x} \mid \tilde{\boldsymbol{x}}) \neq \pi(\tilde{\boldsymbol{x}})q(\tilde{\boldsymbol{x}} \mid \boldsymbol{x})$. Suppose without loss of generality that a pair $(\tilde{\boldsymbol{x}}, \boldsymbol{x})$ satisfies $\pi(\boldsymbol{x})q(\boldsymbol{x} \mid \tilde{\boldsymbol{x}}) > \pi(\tilde{\boldsymbol{x}})q(\tilde{\boldsymbol{x}} \mid \boldsymbol{x})$. Then we include the multiplicative terms $0 < \alpha(\boldsymbol{x} \mid \tilde{\boldsymbol{x}}) < 1$ and $\alpha(\tilde{\boldsymbol{x}} \mid \boldsymbol{x})$ to form a new transition probability $p(\tilde{\boldsymbol{x}} \mid \boldsymbol{x}) \propto q(\tilde{\boldsymbol{x}} \mid \boldsymbol{x})\alpha(\tilde{\boldsymbol{x}} \mid \boldsymbol{x})$ under which the pair $(\tilde{\boldsymbol{x}}, \boldsymbol{x})$ satisfies

$$\pi(\tilde{\boldsymbol{x}})\underbrace{q(\boldsymbol{x} \mid \tilde{\boldsymbol{x}})\alpha(\boldsymbol{x} \mid \tilde{\boldsymbol{x}})}_{p(\boldsymbol{x}|\tilde{\boldsymbol{x}})} = \pi(\boldsymbol{x})\underbrace{q(\tilde{\boldsymbol{x}} \mid \boldsymbol{x})\alpha(\tilde{\boldsymbol{x}} \mid \boldsymbol{x})}_{p(\tilde{\boldsymbol{x}}|\boldsymbol{x})}. \quad (1.3)$$

Analogously, for pairs $(\tilde{\boldsymbol{x}}, \boldsymbol{x})$ satisfying $\pi(\tilde{\boldsymbol{x}})q(\boldsymbol{x} \mid \tilde{\boldsymbol{x}}) < \boldsymbol{\pi}(\boldsymbol{x})q(\tilde{\boldsymbol{x}} \mid \boldsymbol{x})$, we take $0 < \alpha(\tilde{\boldsymbol{x}} \mid \boldsymbol{x}) < 1$ and $\alpha(\boldsymbol{x} \mid \tilde{\boldsymbol{x}})$ where $\alpha(\boldsymbol{x} \mid \tilde{\boldsymbol{x}})$ is also chosen to satisfy equation 1.3. We can combine both cases by taking

$$\alpha(\boldsymbol{x} \mid \tilde{\boldsymbol{x}}) = \left\{1,\ \frac{\pi(\boldsymbol{x})q(\tilde{\boldsymbol{x}} \mid \boldsymbol{x})}{\pi(\tilde{\boldsymbol{x}})q(\boldsymbol{x} \mid \tilde{\boldsymbol{x}})}\right\}.$$

The chain with transition $p(\tilde{\boldsymbol{x}} \mid \boldsymbol{x}) \propto \alpha(\tilde{\boldsymbol{x}} \mid \boldsymbol{x})q(\tilde{\boldsymbol{x}} \mid \boldsymbol{x})$ satisfies the detailed balance condition. Notice that $\alpha(\tilde{\boldsymbol{x}} \mid \boldsymbol{x})$ can be evaluated even if we only have the kernel of the target distribution available analytically.

We iteratively sample from the Markov chain with transition probability $p(\tilde{\boldsymbol{x}} \mid \boldsymbol{x}) = \alpha(\tilde{\boldsymbol{x}} \mid \boldsymbol{x})q(\tilde{\boldsymbol{x}} \mid \boldsymbol{x}) + (1 - \alpha)I_{\boldsymbol{x}}(\tilde{\boldsymbol{x}})$ by first proposing a new value $\boldsymbol{x}^*$ from the proposed transition at the current state $q(\boldsymbol{x}^* \mid \boldsymbol{x})$. We accept $\boldsymbol{x}^*$ with probability $\alpha(\boldsymbol{x}^* \mid \boldsymbol{x})$. If it is accepted, we make $\tilde{\boldsymbol{x}} = \boldsymbol{x}^*$, otherwise we take $\tilde{\boldsymbol{x}} = \boldsymbol{x}$.

In the context of Bayesian inference, the target invariant distribution is the posteriori $\pi(\boldsymbol{x}) = \pi(\boldsymbol{\theta} \mid \boldsymbol{y})$ with $\mathcal{X} = \Theta$ and the acceptance probability simplifies to

$$\alpha(\tilde{\boldsymbol{\theta}} \mid \boldsymbol{\theta}) = \left\{1,\ \frac{p(\boldsymbol{y} \mid \boldsymbol{\theta})\pi(\boldsymbol{\theta})q(\tilde{\boldsymbol{\theta}} \mid \boldsymbol{\theta})}{p(\boldsymbol{y} \mid \tilde{\boldsymbol{\theta}})\pi(\tilde{\boldsymbol{\theta}})q(\boldsymbol{\theta} \mid \tilde{\boldsymbol{\theta}})}\right\}.$$

Pseudocode for the implementation of a Metropolis Hastings transition probability in the context of Bayesian inference is presented in Algorithm 1. Assuming that the Markov chain is ergodic, the process $\hat{\Theta} := \{\boldsymbol{\theta}_{(i)} :\ i = 1, \ldots, M\}$ in Algorithm 1 provides an (approximate) Monte Carlo sample from $\pi(\boldsymbol{\theta} \mid \boldsymbol{y})$. See for example Robert and Casella (2013) for details. Averages over $\hat{\Theta}$ provide the desired approximation of integrals with respect to the target

$\pi(\boldsymbol{\theta} \mid \boldsymbol{y})$.

Algorithm 1: Metropolis Hastings algorithm for posterior samples

Initialize $\boldsymbol{x}(1) \sim \pi_0(\boldsymbol{\theta})$;
Choose the proposal $q(\tilde{\boldsymbol{\theta}} \mid \boldsymbol{\theta})$ (irreducible and aperiodic);
for $(i \leq M)$ **do**
  Propose $\tilde{\boldsymbol{\theta}} \sim q(\tilde{\boldsymbol{\theta}} \mid \boldsymbol{\theta}_{(i)})$;
  Calculate $\alpha(\tilde{\boldsymbol{\theta}}, \boldsymbol{\theta}_{(i)}) = \left\{1, \ \frac{p(\boldsymbol{y}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})q(\tilde{\boldsymbol{\theta}}|\boldsymbol{\theta})}{p(\boldsymbol{y}|\tilde{\boldsymbol{\theta}})\pi(\tilde{\boldsymbol{\theta}})q(\boldsymbol{\theta}|\tilde{\boldsymbol{\theta}})}\right\}$;
  Sample $u_{(i)} \sim Unif(0, 1)$;
  if $u_{(i)} \leq \alpha(\tilde{\boldsymbol{\theta}} \mid \boldsymbol{\theta})$ **then**
    $\boldsymbol{\theta}_{(i+1)} \leftarrow \tilde{\boldsymbol{\theta}}$;
  else
    $\boldsymbol{\theta}_{(i+1)} \leftarrow \boldsymbol{\theta}_{(i)}$;
  end
end

Possible choices for the proposal $q(\tilde{\boldsymbol{\theta}} \mid \boldsymbol{\theta})$ are

1. Independent proposal: $q(\tilde{\boldsymbol{\theta}} \mid \boldsymbol{\theta}) = q(\tilde{\boldsymbol{\theta}}) \ \forall \ \boldsymbol{\theta}, \tilde{\boldsymbol{\theta}} \in \Theta$. The independent proposal does not depend on the current state of the chain. The closer $q(\tilde{\boldsymbol{\theta}})$ is from $\pi(\tilde{\boldsymbol{\theta}} \mid \boldsymbol{y})$, the higher the chance of accepting proposed values which defines a better mixing Markov chain.

2. Random walk proposal: $q(\tilde{\boldsymbol{\theta}} \mid \boldsymbol{\theta}) = q(\boldsymbol{\theta} \mid \tilde{\boldsymbol{\theta}})$, e.g., $q(\tilde{\boldsymbol{x}} \mid \boldsymbol{x}) = N(\tilde{\boldsymbol{\theta}} \mid \boldsymbol{\theta}; \boldsymbol{V})$ for the tunning covariance matrix $\boldsymbol{V}$. Typically, we take $\boldsymbol{V} =$

$\text{diag}(v_1^2, \ldots, v_p^2)$ to be a diagonal matrix. We propose a new value centered on the current one. For component k, if v_k^2 is too big, then the proposal is too erratic, leading to a low acceptance probability. On the other hand, for values of v_k^2 too small we get a chain with very high acceptance, but moving too slowly in each iteration, i.e. a slowly mixing Markov chain. Therefore, some tunning of v_k^2 is usually necessary.

Under both proposals, a sufficient condition for an irreducible and aperiodic chain p is $P_q(\tilde{\boldsymbol{\theta}} \in A \mid \boldsymbol{\theta}) > 0 \quad \forall A \subset \Theta$ measurable (meaning that q allows the chain to move to any measurable set within the support Θ within a single move).

1.4.2 Gibbs sampler

Consider $\boldsymbol{x} = (x_1, \ldots, x_p) \in \mathbb{R}^p$ and the target distribution $\pi(\boldsymbol{x})$ again analytically available up to an unknown multiplicative normalization constant. The Gibbs sampler operates by sequentially sampling from the full conditional distributions $\pi(x_k \mid \boldsymbol{x}_{-k}),\ k = 1, \ldots, p$, as stated in algorithm 2 where the target is, again, the posterior distribution: $\pi(\boldsymbol{x}) = \pi(\boldsymbol{\theta} \mid \boldsymbol{y})$.

Algorithm 2: Gibbs sampling algorithm for posterior samples

Initialize $\boldsymbol{\theta}^{(1)} \sim \pi_0(\boldsymbol{\theta})$;
for $(i \leq M)$ **do**
  Sample component 1: $\theta_1^{(i+1)} \sim \pi(\theta_1 \mid \boldsymbol{y}, \theta_2^{(i)}, \ldots, \theta_p^{(i)})$;
  Sample component 2: $\theta_2^{(i+1)} \sim \pi(\theta_2 \mid \boldsymbol{y}, \theta_1^{(i+1)}, \theta_3^{(i)}, \ldots, \theta_p^{(i)})$;
  $\vdots$
  Sample component p: $\theta_p^{(i+1)} \sim \pi(\theta_p \mid \boldsymbol{y}, \theta_1^{(i+1)}, \ldots, \theta_{p-1}^{(i+1)})$;
end

1.4.3 Reversible jumps and variable dimensions

Many BNP models involve parameter vectors of variable dimension. In order to accomodate for this, it is common to extend the Metropolis-Hastings algorithm to propose transdimensional moves using a reversible jump MCMC (Green, 1995). In this section, we provide a brief summary of reversible jump MCMC (RJMCMC). More details and examples are available in Green (1995) and Richardson and Green (1997).

In the following discussion, let $\boldsymbol{x}$ denote the parameter vector. The target distribution is denoted by $\pi(\boldsymbol{x})$ for transdimensional $\boldsymbol{x} \in \cup_{n\in\mathbb{N}}\mathbb{R}^n$. The target distribution restricted to $\mathbb{R}^n$ is denoted by $\pi_n(\boldsymbol{x}_n)$ and it has density $f_n(\boldsymbol{x}_n)$. We start defining up and down moves that will respectively increase or decrease the dimensionality of the parameter. As always in a Markov chain, transition probabilities are allowed to depend on the current state; for example, down moves in a mixture model could be proposed by selecting which pair of the current components (clusters) to merge. For a state $\boldsymbol{x} \in \mathbb{R}^n$, the list of all (finite) possible up and down moves are $M_u(\boldsymbol{x}) = \{u_1(\boldsymbol{x}), \ldots, u_{n_x}(\boldsymbol{x})\}$ and $M_d(\boldsymbol{x}) = \{d_1(\boldsymbol{x}), \ldots, d_{n_x}(\boldsymbol{x})\}$ respectively. We will denote $M(\boldsymbol{x}) = M_d(\boldsymbol{x}) \cup M_u(\boldsymbol{x})$.

Finally, let $q_m(\boldsymbol{x})$ be the probability of using the transition probability $m \in M(\boldsymbol{x})$ when the current state is $\boldsymbol{x}$.

Furthermore, suppose all up moves $u \in M_u(\boldsymbol{x})$ from any state $\boldsymbol{x}$ to a state $\boldsymbol{y}$ are obtained by sampling auxiliary variables $\boldsymbol{v} \sim q_{aux}(\boldsymbol{v})$ and then

applying the deterministic invertible transformation $y = T_u(\boldsymbol{x}, \boldsymbol{v})$. Notice that given the current state $\boldsymbol{x}$ and the up move u, the proposed value $\boldsymbol{y} = T_u(\boldsymbol{x}, \boldsymbol{v})$ is random due to the randomness of $\boldsymbol{v}$. On the other hand, we will assume that proposed values from down moves $d \in M(\boldsymbol{y})$ are obtained deterministically, given the current state $\boldsymbol{y}$ and the down move d. One last definition: $\alpha_m(\boldsymbol{x}, \boldsymbol{y})$ is the probability of accepting the proposed value of $\boldsymbol{y}$ given the transition probability $m \in M(\boldsymbol{x})$ and the current state $\boldsymbol{x}$. Notice that $\alpha_m(\boldsymbol{x}, \boldsymbol{y})$ depends on the auxiliary variable $\boldsymbol{v}$.

Let where $|J| = \det(\partial T / \partial \boldsymbol{x} \partial \boldsymbol{v})$ denote the Jacobian of transformation T. Finally, the reversible jump MCMC uses the following acceptance probabilities for up and down moves:

$$\alpha_u(\boldsymbol{x}, \boldsymbol{y}) = \min\left\{1, \frac{q_d(\boldsymbol{y}) f_{n+1}(\boldsymbol{y}) |J|}{q_u(\boldsymbol{x}) q_{aux}(\boldsymbol{v}) f_n(\boldsymbol{x})}\right\},$$
$$\alpha_d(\boldsymbol{y}, \boldsymbol{x}) = \min\left\{1, \frac{q_u(\boldsymbol{x}) q_{aux}(\boldsymbol{v}) f_n(\boldsymbol{x})}{q_d(\boldsymbol{y}) f_{n+1}(\boldsymbol{y}) |J|}\right\} = \min\left\{1, \frac{1}{\alpha_u(\boldsymbol{x}, \boldsymbol{y})}\right\}.$$

It can be shown that the defined transition probabilities satisfy the detailed balance condition.

Chapter 2

A Bayesian Random Partition Model for Sequential Refinement and Coagulation

2.1 Introduction

This work has appeared in Zanini et al. (2019).

2.1.1 Overview

In this section, we propose a model for a sequence of partitions that includes refinement of the initial partition followed by later coagulation. The model is motivated by an analysis of protein activation over time after an intervention.

A functional protein pathway involves proteins whose expressions are dependent. For example, expression of a protein can stimulate the expression of another protein. Usually a stable pathway leads to an equilibrium state of the expression of all proteins in the pathway, which can be modeled as a probability distribution. In cancer cells, biological pathways are almost always disrupted, which shifts the equilibrium state of the protein expression. Effective cancer drugs, such as targeted protein inhibitors, can help treat cancer patients by altering protein expression for key biomarkers. Through pathways, other proteins are subsequently affected which ultimately leads to phenotypes

that are beneficial for patient survival or quality of life. For a new developmental drug, one of the first steps is to test which proteins are affected when the drug is introduced to the cells. This is typically done by functional assays. We consider such an assay in which protein expression of a biological pathway is measured at the baseline and at multiple time points after a drug is introduced to cancer cells.

We analyze protein expression data from such functional assays. To investigate which proteins have their expression levels changed (directly or indirectly) after being exposed to the drug, we define a Bayesian model for protein expressions with a time-dependent clustering structure. The underlying assumption of the model is a stylized representation of the earlier description of disrupted protein pathways. We assume that the proteins are originally clustered in a canonical way with respect to their protein expressions and, after a certain time period of drug exposure, some or all of the proteins have their expressions altered, which may lead to a different clustering structure. As time passes the drug effect wears off, and the clustering structure of the proteins may revert to the initial state. In other words, we model protein expression and the treatment effect by arranging proteins in different subsets (clusters), possibly corresponding to biologic function, with cluster-specific mean expression levels. Treatment response is modeled by allowing a change in cluster-specific means over a time interval after treatment, and by adding new clusters to allow for heterogeneous treatment responses. The model includes random time points to define this time interval after treat-

ment. Methodologically, through a Bayesian modeling framework we propose an approach that allows inference for such dependent and temporal clustering. The dependence is on the partitions that define the clusters, rather than on the distribution of these partitions.

The proposed process is a reduced and simplified version of the more general fragmentation and coagulation process of Teh et al. (2011). Another more general model, without explicit modeling of refinement or coagulation, is proposed in Elliott et al. (2018) who use a hierarchical Dirichlet process to infer local genetic ancestry from genotype data. The model implies a partition of subjects into subsets with common ancestry at each locus. Partitions are allowed to vary across genetic locus and dependence is formalized by a hidden Markov process. In general, any dependent discrete random probability model such as the Dirichlet or Pitman-Yor (PY) processes (Pitman and Yor, 1987), indexed by discrete time, could be used to induce the desired time-dependent random partition. Such models are developed in Caron et al. (2017) and Rodríguez and Ter Horst (2008). Caron et al. (2017) construct a sequence of random partitions with each partition marginally distributed as in a PY mixture model, with an additional parameter to control similarity between partitions. The approach is based on a property of the Ewens sampling formula known as consistency under deletion (Kingman, 1978). However, these models for sequences of random partitions are more general than what is needed here and the implied marginal distribution of the random partition at each time point is the same. In contrast, the assumed monotonicity of

fragmentation and following coagulation is important in our application. It represents how the treatment affects the proteins (refinement), and that effect eventually vanishes (coagulation). This desired monotonicity (of adding and then removing clusters) and the limited data in the motivating application lead us to construct a much simplified version of such more general models. The main inference target is the subset of proteins that form the refined partition clusters, corresponding to the desired subset of proteins that are most affected by the initial treatment.

2.1.2 Dataset

The motivating data are from an experiment using reverse phase protein arrays (RPPA) which record the expression of selected proteins in a biological pathway simultaneously on multiple samples. Multiple cell line samples are prepared and exposed to multiple protein inhibitors at different dose levels (Charboneau et al., 2002). The experimental design is a balanced factorial structure, including $C = 2$ cell lines, $D = 3$ drugs, $J = 3$ technical replicates, and $L = 4$ doses (0, 0.625, 2.5 and 10uM), with expression measurements of $I = 55$ proteins recorded at $T = 8$ different times $(0, 5, 15, 30, 60, 90, 120$ and 180 minutes) after the drug exposure. The cell cultures are treated with three protein inhibitors that are often investigated in cancer studies. The included drugs act on: Phosphoinositide 3-Kinase (PI3K), which is responsible for coordinating cell functions such as proliferation, cell survival, degranulation, vesicular trafficking and cell migration (Azadi et al., 2016); Protein

Kinase B (AKT), which promotes growth factor-mediated cell survival, cell proliferation and inhibits apoptosis through the inactivation of pro-apoptotic proteins (Nitulescu et al., 2016); and mitogen-activated protein kinase kinase (MEK), which is an important component of the ERK1/2 signaling pathway that is often deregulated in cancer cells (Caunt et al., 2015).

Some of the data can be seen in the four panels in the left column of Figure 2.7. The plots show the data for cell line $c = 1$ under drug $d = 1$, which is the PI3K inhibitor. The horizontal axis is time (in minutes) after treatment. The vertical axis is protein expression (averaged over $J = 3$ repeat experiments). Notice how some proteins have their expressions altered after the dose is administered. Figure 2.8 shows the same for cell line $c = 2$.

For an initial exploratory data analysis one could use a fit of the trajectory for each protein and try to identify systematic changes. Figure 2.1, for example, shows a fit of the data using a flexible regression model – in this case smoothing splines. While the fit is reasonable, it remains difficult to spot proteins that respond to treatment. By arranging proteins in clusters we will reduce some of the noise and be able to highlight possible treatment effects.

2.2 Probability Model

Let $y_{cd\ell ijt}$ denote the expression level for protein i in cell line c, drug d, dose ℓ, replicate j and time point t. To simplify notation, we drop c and d from the subindex in the following discussion as they appear in (almost) all

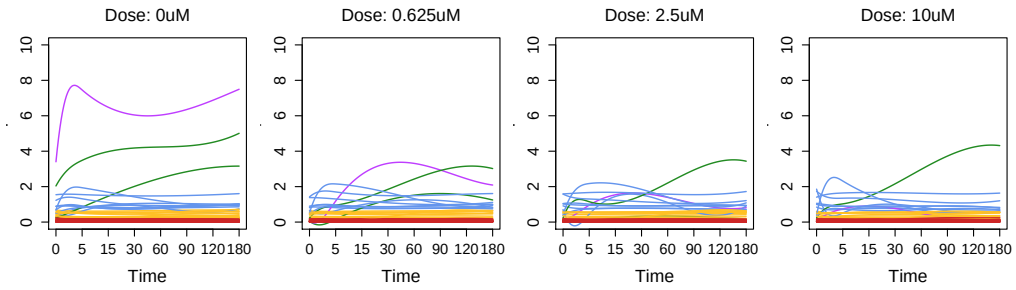


Figure 2.1: Smoothing splines based on the J=3 repetitions over time (minutes) for each protein in cell line 1 for drug PI3Ki. Cluster structure (represented in colors) is constrained to be the same for all different doses. Compare with the data shown in the left column of Figure 5.

variables. Only one parameter, $\boldsymbol{\Sigma}_c$ is common across drugs which we highlight by including the c subindex for $\boldsymbol{\Sigma}_c$. In addition, some of the hyperparameters are common across all c, d as indicated below. We use notation for column vectors such as $(a_n)_{n=1}^N = (a_1, \ldots, a_N)^\top$.

We assume a model $y_{\ell ijt} = \mu_{\ell it} + \epsilon_{\ell ijt}$, where $\mu_{\ell it}$ is the mean expression level for a specific cell line, drug, dose, protein and time, and $\epsilon_{\ell ijt}$ represent time-dependent Gaussian errors. Let $\boldsymbol{\epsilon}_{\ell ij} = (\epsilon_{\ell ijt})_{t=1}^T$ denote the error vector. We assume $\boldsymbol{\epsilon}_{\ell ij} \overset{\text{iid}}{\sim} N(\mathbf{0}, \boldsymbol{\Sigma}_\mathbf{c})$ with $\boldsymbol{\Sigma}_c$ denoting covariance matrix, independently across cell line, drugs, doses, proteins and replicates. Similarly let $\boldsymbol{y}_{\ell ij} = (y_{\ell ijt})_{t=1}^T$ and $\boldsymbol{\mu}_{\ell i} = (\mu_{\ell it})_{t=1}^T$. The joint likelihood becomes

$$\boldsymbol{y}_{\ell ij} \overset{\text{ind}}{\sim} N(\boldsymbol{\mu}_{\ell i}, \boldsymbol{\Sigma}_c), \tag{2.1}$$

with independence across all subindex values, but dependence of the elements $y_{\ell ijt}$ across time $t = 1, \ldots, T$. We assume an inverse Wishart conjugate prior on the covariance matrices $\boldsymbol{\Sigma}_c \overset{\text{ind}}{\sim} \text{IW}(\nu_\Sigma, \boldsymbol{V}_\Sigma)$, $c = 1, \ldots, C$, with expectation

$\frac{\boldsymbol{V}_\Sigma}{\nu_\Sigma - T - 1}$. Here $\boldsymbol{V}_\Sigma$ is a (fixed) $T \times T$ matrix-variate hyperparameter and $\nu_\Sigma \geq T + 2$ are the degrees of freedom. Introducing a more detailed model for temporal dynamics is not meaningfully possible with the small sample sizes and only $T = 8$ longitudinal observations.

We introduce a time-dependent partition of the proteins, which together with cluster-specific means implies a generative model for the mean protein expressions $\boldsymbol{\mu}_{\ell i}$ within cell line, drug and dose, and across different time points $t = 1, \ldots, T$. We first develop the structure for the time-dependent partitions. Let $\boldsymbol{\delta}_t = (\delta_{ti})_{i=1}^I$ denote the partition at time t of proteins $i = 1, \ldots, I$ into κ^t clusters $m = 1, \ldots, \kappa^t$. The random partitions $\boldsymbol{\delta}_t$ are characterized by cluster membership indicators $\delta_{ti} \in \{1, \ldots, \kappa^t\}$ with $\delta_{ti} = m$ when protein i is in the m-th cluster at time t. A key model feature is the prior probability model for the sequence of partitions $\boldsymbol{\delta}_1, \ldots, \boldsymbol{\delta}_T$ that defines the evolution of the partitions over time (as before, separately for each cell-line c and drug d). See below for the choice of κ^t – we will reduce it to only two distinct values, κ_1 and κ_2, over time. Also, for clarification we note that the dependence should be on the partitions themselves rather than on their distributions. Modeling the dependence on the actual proteins, i.e., the cluster membership of proteins, allows us to represent how the treatment affects each protein.

One desired feature motivated by the nature of the RPPA data analysis is that partitions should initially start fragmenting (i.e., more subsets should be formed) up to a certain change point, after which a coagulation process starts (i.e., merging of clusters into fewer subsets). This reflects the drug

action on proteins. In other words, the drug is expected to alter the regular expression pattern of the proteins, resulting in more heterogeneous expression profiles and therefore more clusters. As the drug effect wears off, the expression of the proteins should revert to the original states, implying a coagulation of the clusters.

We implement the desired structure with two change-points in time. The first change point marks the beginning of the refined partition with more clusters, and the second change point marks the time when the partition reverts to the original clusters. We let τ_ℓ^1 (*refinement* change point) denote the first change point when the proteins form the finer partition, and let τ_ℓ^2 (*coagulation* change point) denote the second change point. We assume $1 \leq \tau_\ell^1 < \tau_\ell^2 \leq T$ for all cell lines c, drugs d, and doses ℓ. One key feature is that τ_ℓ^1 and τ_ℓ^2 are specific to dose ℓ. This represents how different doses act faster or slower on the proteins. Higher doses are expected to start acting on the proteins earlier than lower doses, i.e., we expect monotonicity of τ_ℓ^1 and τ_ℓ^2 over doses. We further assume $(\tau_\ell^1, \tau_\ell^2) \overset{\text{iid}}{\sim} \text{Unif}\left(\{(u_1, u_2) : 1 \leq u_1 < u_2 \leq T-1\}\right)$. Adding an informative prior would be straightforward. However, even with the (vague) uniform prior we find little posterior uncertainty on the change points.

The prior probability models for the baseline and fragmented partitions are constructed as Dirichlet-multinomial models for cluster membership indicators. Let $\boldsymbol{\delta}^u = (\delta_i^u)_{i=1}^I$ for $u \in \{1, 2\}$ denote the two partitions of proteins with $u = 1$ indicating the original (coarse) partition that applies for $t < \tau_\ell^1$ and $t > \tau_\ell^2$, and $u = 2$ indicating the (refined) partition that applies for

$\tau_\ell^1 \leq t \leq \tau_\ell^2$. That is,

$$\boldsymbol{\delta}_t = \begin{cases} \boldsymbol{\delta}^1, \text{ for } 1 \leq t \leq \tau_\ell^1, \text{ or } t \geq \tau_\ell^2 + 1 \\ \boldsymbol{\delta}^2, \text{ for } \tau_\ell^1 + 1 \leq t \leq \tau_\ell^2. \end{cases}$$

We assume that $P(\delta_i^1 = m) = \pi_m^1$ for clusters $m = 1, \ldots, \kappa_1$. The prior for the fragmented partition $\boldsymbol{\delta}^2$ is constructed in two steps. First, set $\delta_i^2 = \delta_i^1$ with probability γ; second, all proteins i with $\delta_i^2 \neq \delta_i^1$ are gathered in the set $\mathcal{A} := \{i : \ \delta_i^1 \neq \delta_i^2\}$, and form new clusters by $P(\delta_i^2 = m) = \pi_{m-\kappa_1}^2$, $m = \kappa_1 + 1, \ldots, \kappa_2, \ i \in \mathcal{A}$. Note that $p(\boldsymbol{\delta}^2 \mid \boldsymbol{\delta}^1)$ does not define $\boldsymbol{\delta}^2$ as a partition nested within $\boldsymbol{\delta}^1$. This is why we use the term refinement throughout.

We assume independent priors for γ, $\boldsymbol{\pi}_1 = (\pi_1^1, \ldots, \pi_{\kappa_1}^1)$ and $\boldsymbol{\pi}_2 = (\pi_1^2, \ldots, \pi_{\kappa_2 - \kappa_1}^2)$ as $\gamma \sim \text{Beta}(a_\gamma, b_\gamma)$ $\boldsymbol{\pi}_1 \sim \text{Dir}(\boldsymbol{\eta}_1)$, and $\boldsymbol{\pi}_2 \sim \text{Dir}(\boldsymbol{\eta}_2)$. The hyperparameters $\gamma, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2$ are shared across all cell lines c and drugs d.

Next, we construct a prior for the mean protein expression $\boldsymbol{\mu}_{\ell i}$ in (2.1) by defining cluster-specific common values. That is, the partition is linked with the protein mean expression. Given $\{\boldsymbol{\delta}_t : \ 1 \leq t \leq T\}$ we assume

$$\mu_{\ell i t} = \begin{cases} \mu_{\ell 1}^*(\delta_i^1), \text{ if } 1 \leq t \leq \tau_\ell^1 \\ \mu_{\ell 2}^*(\delta_i^2), \text{ if } \tau_\ell^1 + 1 \leq t \leq \tau_\ell^2 \\ \mu_{\ell 3}^*(\delta_i^1), \text{ if } \tau_\ell^2 + 1 \leq t. \end{cases} \tag{2.2}$$

In words, $\mu_{\ell i t} = \mu_{\ell u}^*(m)$ for all proteins in cluster m under dose level ℓ in the time interval $u = 1, 2$ or 3, with the time intervals corresponding to the initial, fragmented and final partitions respectively (initial and final partitions are assumed equal). The choice of the piecewise constant mean function in (2.2) is only for parsimony. Alternatively, one could use a piecewise linear mean response, without much change in the remaining discussion.

The model is completed with a prior on the cluster-specific parameters, $(\mu^*_{\ell u}(m) \mid \mu_{0u}, v_{0u}) \overset{\text{iid}}{\sim} N(\mu_{0u}, v_{0u}^{-1})$, $\mu_{0u} \overset{\text{iid}}{\sim} N(\mu_{00}, v_{00}^{-1})$ and $v_{0u} \overset{\text{iid}}{\sim} \text{Gamma}(a_v, b_v)$. The hyperparameters (μ_{0u}, v_{0u}) are common across cell lines c and drugs d.

In summary, the proposed model constructs a mixture of Gaussian sampling model for the observed protein expressions over time, with the mixture being induced by the latent partitions $\boldsymbol{\delta}^1$ and $\boldsymbol{\delta}^2$. In fact, marginalizing $\boldsymbol{\delta}^1$ and $\boldsymbol{\delta}^2$, we find the following mixture of normals sampling model. Let $\boldsymbol{u}_1 = (1, \ldots, 1, 0, \ldots, 0)^\top$, $\boldsymbol{u}_2 = (0, \ldots, 0, 1, \ldots, 1, 0, \ldots, 0)^\top$ and $\boldsymbol{u}_3 = (0, \ldots, 0, 1, \ldots, 1)^\top$ denote design vectors with 1's in positions $1, \ldots, \tau^1_\ell$ (for $\boldsymbol{u}_1$), in positions $\tau^1_\ell + 1, \ldots, \tau^2_\ell$ (for $\boldsymbol{u}_2$) and in positions $\tau^2_\ell + 1, \ldots, T$ (for $\boldsymbol{u}_3$), respectively, and let $\boldsymbol{\mu}^*_\ell(k_1, k_2) = \mu^*_{\ell 1}(k_1)\, \boldsymbol{u}_1 + \mu^*_{\ell 2}(k_2)\, \boldsymbol{u}_2 + \mu^*_{\ell 3}(k_1)\, \boldsymbol{u}_3$ denote the T-dimensional mean vector for proteins in clusters k_1 and k_2 under the initial and the refined partition, respectively. Let $N(\boldsymbol{x}; \boldsymbol{m}, \boldsymbol{S})$ denote a multivariate normal p.d.f. evaluated at $\boldsymbol{x}$ with mean $\boldsymbol{m}$ and covariance matrix $\boldsymbol{S}$. Then

$$
\begin{aligned}
p(\boldsymbol{y}_{\ell ij} \mid \tau^1_\ell, \tau^2_\ell, \boldsymbol{\Sigma}_c, \boldsymbol{\mu}^*_{\ell 1}, \boldsymbol{\mu}^*_{\ell 2}, \boldsymbol{\mu}^*_{\ell 3}, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \gamma) = \\
= (1-\gamma) \sum_{k_1=1}^{\kappa_1} \sum_{k_2=1}^{\kappa_2 - \kappa_1} \pi^1_{k_1} \pi^2_{k_2} N(\boldsymbol{y}_{\ell ij}; \boldsymbol{\mu}^*_\ell(k_1, k_2), \boldsymbol{\Sigma}_c) + \\
+ \gamma \sum_{k_1=1}^{\kappa_1} \pi^1_{k_1} N(\boldsymbol{y}_{\ell ij}; \boldsymbol{\mu}^*_\ell(k_1, k_1), \boldsymbol{\Sigma}_c).
\end{aligned}
$$

Note how the proposed model is different from models that allow dependence of the distributions for the random partitions. Dependence in the prior on the random partitions over time would not necessarily enforce the

desired monotonicity of refinement (to represent the treatment effect) and following coagulation for an actual realization of protein-specific cluster membership. Here, the dependence is built on the partitions themselves, unlike, for example, the earlier mentioned model of Caron et al. (2017) where each $\boldsymbol{\delta}_t$ is marginally distributed according to a PY-style (Generalized Pólya urn) distribution, exploring several ways to relate and control similarity across partitions. The fragmentation and coagulation feature cannot be represented by models with invariant marginal distribution. Modeling the desired monotone pattern of change in the partition is the key motivation for the proposed construction.

Finally, we would like to comment on the choice of the proposed model versus a seemingly simpler parametric model, such as a linear mixed effects model or a regression with splines, as in Figure 1 in the supporting information section. While such parametric models could adequately model time-dependent mean response, inference for protein-specific response to treatment effects would require corresponding protein- and time-specific random effects.

2.3 Posterior Inference

We implement Markov chain Monte Carlo (MCMC) posterior simulation. Let $\boldsymbol{\theta}$ denote the complete parameter vector. For posterior simulation it is now important to keep track of parameters that are in common across cell lines c and drugs d. We therefore start to include the subindexes c and d again

as needed. The joint prior distribution can be factorized as

$$p(\boldsymbol{\theta}) \propto \left\{\prod_{u=1}^{3} p(v_{0u})\right\}\left\{\prod_{u=1}^{3} p(\mu_{0u})\right\}\left\{\prod_{c=1}^{C}\prod_{d=1}^{D}\prod_{\ell=1}^{L} p(\tau^1_{cd\ell}, \tau^2_{cd\ell})\right\} p(\gamma)$$
$$\times \left\{\prod_{c=1}^{C}\prod_{d=1}^{D} p(\boldsymbol{\delta}^1_{cd} \mid \boldsymbol{\pi}_1) p(\boldsymbol{\delta}^2_{cd} \mid \boldsymbol{\delta}^1_{cd}, \gamma, \boldsymbol{\pi}_2)\right\} \times p(\boldsymbol{\pi}_1)p(\boldsymbol{\pi}_2) \times \prod_{d=1}^{D}\prod_{c=1}^{C} p(\boldsymbol{\Sigma}_c)$$
$$\times \prod_{c=1}^{C}\prod_{d=1}^{D}\prod_{\ell=1}^{L}\left\{\underbrace{\prod_{m=1}^{\kappa_1} p(\mu^*_{cd\ell 1}(m) \mid \mu_{01}, v_{01})}_{u=1} \times \underbrace{\prod_{m=1}^{\kappa_2} p(\mu^*_{cd\ell 2}(m) \mid \mu_{02}, v_{02})}_{u=2} \times \right.$$
$$\left. \times \underbrace{\prod_{m=1}^{\kappa_1} p(\mu^*_{cd\ell 3}(m) \mid \mu_{03}, v_{03})}_{u=3}\right\},$$

where κ_1 is the number of clusters in time intervals $\{t : \ 1 \leq t \leq \tau^1_{cd\ell}\}$ (corresponding to $u = 1$) and $\{t : \ t > \tau^2_{cd\ell}\}$ (or $u = 3$); and κ_2 is the number of clusters in time interval $\{t : \ \tau^1_{cd\ell} + 1 \leq t \leq \tau^2_{cd\ell}\}$ (or $u = 2$). If desired, the model could easily be generalized to different number of clusters across cell line and drugs. The likelihood is given by the independent normal sampling model

$$\prod_{c=1}^{C}\prod_{d=1}^{D}\prod_{i=1}^{I}\prod_{\ell=1}^{L}\prod_{j=1}^{J} N(\boldsymbol{y}_{cd\ell ij};\ \boldsymbol{\mu}_{cd\ell i},\ \boldsymbol{\Sigma}_c).$$

Although posterior inference is not analytically tractable for this model, conditional conjugacy implies that all full conditionals are well known distributions that are straightforward to sample from (see Web Appendix A). We therefore implement MCMC simulation using a Gibbs sampler Markov chain. We run one common Markov chain for inference across all (c, d), but report inference

on partitions separately for each (c, d). Therefore, in the following discussion of inference summaries, we drop the $_{cd}$ subindex again.

Point estimates of the cluster-membership indicators are obtained using the approach proposed by Dahl (2006). After judging (practical) convergence of the MCMC algorithm, we evaluate for each pair $i < j$ of proteins, the pairwise co-clustering probability $\hat{p}_{ij} = \frac{1}{K}\sum_k p_{ij}^{(k)}$, where K is the Monte Carlo sample size and $p_{ij}^{(k)}$ is an indicator for i and j being allocated to the same cluster. The $p_{ij}^{(k)}$ and $\hat{p}_{ij}$ are combined into $(I \times I)$ matrices $P^{(k)} = [p_{ij}^{(k)}]$ and $\hat{\boldsymbol{P}} = [\hat{p}_{ij}]$. We then report as posterior estimated $\bar{\boldsymbol{\delta}}$ the partition corresponding to the $\boldsymbol{P}^{(k^*)}$ that minimizes $||\hat{\boldsymbol{P}} - \boldsymbol{P}^{(k)}||$, i.e., $k^\star = \arg\min_k ||\hat{\boldsymbol{P}} - \boldsymbol{P}^{(k)}||$. In words, $k^\star$ indexes the Monte Carlo sample whose co-clustering matrix is closest to $\hat{\boldsymbol{P}}$. Once the point estimate of the clustering structures is obtained, we run a new MCMC chain with fixed cluster membership indicators to carry out inference for the remaining parameters, now conditional on the estimated partition.

Finally, we consider learning about the unknown size κ_1 and κ_2 of the partitions. Using transdimensional transition probabilities, such as reversible jump (Green, 1995), the selection of these parameters could be included in the same MCMC simulation. However, we found that the implementation of such transition probabilities is impractical for the proposed model. Considering a variation of reversible jump for multivariate mixtures of normals with split and merge proposals that are constructed to maintain marginal first and second moments (Zhang et al., 2004; Dellaportas and Papageorgiou, 2006)

we find it impossible to achieve acceptable mixing rates of the Markov chain simulation. The challenge lies in finding a split move that simultaneously proposes reasonable draws for $\mu^*_{cd\ell u}(m)$ across all $\ell \in \{1, \ldots, L\}$, $u \in \{1, 2, 3\}$ and $m \in \{1, \ldots, \kappa_1\}$ with high probability. We therefore recommend to use an alternative model selection framework to determine (κ_1, κ_2). We consider several criteria, including AIC (Akaike, 1973), BIC (Schwarz, 1978), DIC (Spiegelhalter et al., 2002), and WAIC (Watanabe, 2010) as well as log pseudo marginal likelihood (LPML). See, for example, Gelman et al. (2014) for a review on these methods. The specifics of counting the number of parameters, as it is required to evaluate AIC and BIC are described in Web Appendix B. In the following section we report a specific recommendation, based on results in a simulation study.

2.4 Simulation

We carried out several simulation studies to verify that the proposed model allows for meaningful inference in the context of weak signals and relatively small sample sizes as in the RPPA data. We considered two scenarios, with several variations.

Scenario 1: We simulated five hypothetical datasets with the following (true) partition sizes: $(\kappa_1, \kappa_2) \in \{(2, 3), (3, 4), (3, 5), (4, 7), (5, 7)\}$. In all five cases we simulated from the model described in section 2.2. The bottom level hyperparameters were fixed as $\gamma = 0.9$, $v_{0u} = 5$ for $u = 1, 2, 3$ and $\mathbf{\Sigma}_c = 0.1 I_{8\times 8}$, where $I_{8\times 8}$ denotes the 8 dimensional identity matrix. For

any cell line c and dose ℓ, the change points were fixed as $(\tau^1_{cd\ell}, \tau^2_{cd\ell}) = (2, 5)$ for $d = 1$, $(\tau^1_{cd\ell}, \tau^2_{cd\ell}) = (3, 6)$ for $d = 2$ and $(\tau^1_{cd\ell}, \tau^2_{cd\ell}) = (4, 7)$ for $d = 3$. For $(\kappa_1, \kappa_2) = (2, 3), (3, 4), (3, 5)$, we fixed $\mu_{0u} = (0.5, 1.5, 0.4)$, whereas for $(\kappa_1, \kappa_2) = (4, 7), (5, 7)$, we fixed $\mu_{0u} = (1.0, 1.5, 1.4)$. The remaining parameters were randomly generated from the respective prior probability model.

For each dataset we then implement inference in two steps as in section 2.3. First we run 500 MCMC iterations, discarding the first 100 as initial burn-in under each one of 21 possible pairs of (κ_1, κ_2). We use the pairs $\{(a, b) : a \in \{2, \ldots, 8\},\ b \in \{a+1, a+2, a+3\}\ \}$. This first step evaluates the different model choice criteria (see below) to select the best pair (κ_1, κ_2), and then estimates $(\boldsymbol{\delta}^1_{cd}, \boldsymbol{\delta}^2_{cd})$ using the approach of Dahl (2006).

In a second step we simulate 5000 more MCMC iterations to implement inference conditional on the chosen model, i.e., with fixed $\boldsymbol{\delta}^1_{cd}$, $\boldsymbol{\delta}^2_{cd}$, $c = 1, 2,\ d = 1, 2, 3$. The first 2000 iterations are discarded as initial burn-in. Hyperparameters are fixed as $\nu_\Sigma = 10$, $\boldsymbol{V}_\Sigma = I_{8\times 8}$, $a_\gamma = 1$, $b_\gamma = 1$, $\boldsymbol{\eta}_1 = (1, ..., 1)^\top \in \mathbb{R}^{\kappa_1}$, $\boldsymbol{\eta}_2 = (1, ..., 1)^\top \in \mathbb{R}^{\kappa_2 - \kappa_1}$, $\mu_{00} = 0$, $v_{00}^{-1} = 0.4444$, $a_v = 1$ and $b_v = 1$ to reflect weak prior information.

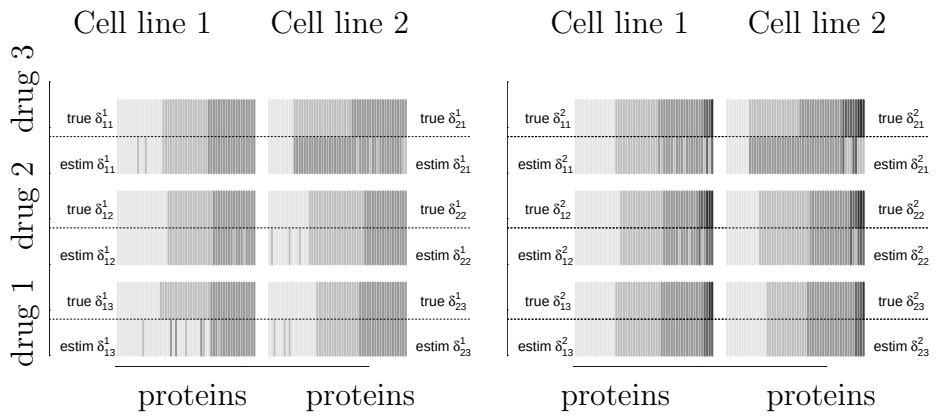
We implement learning about the cluster sizes (κ_1, κ_2) as model selection using various criteria proposed in the literature. We briefly summarize the results in Table 2.1. BIC always selects a more parsimonious model, and AIC, DIC, WAIC and LPML always point to the same model (except under $(5, 7)$). We conclude to use BIC, as it gives the best trade-off of a good fit and selecting parsimonious models.

Table 2.1: Simulation truth and estimate for (κ_1, κ_2) under alternative model selection criteria. * Under simulation truth $(5, 7)$, AIC selects $(5, 6)$.

truth	(2,3)	(3,4)	(3,5)	(4,7)	(5,7)
BIC	(2,5)	(2,5)	(3,4)	(3,6)	(5,6)
AIC, DIC,WAIC, LPML	(4,7)	(3,6)	(5,6)	(4,7)	(8,10)*

Next we summarize results for the simulation scenario with true (κ_1, κ_2) $= (3, 5)$. In this case BIC selects $(\kappa_1, \kappa_2) = (3, 4)$. The objective is to explore whether inference can recover mean parameters and cluster structure for data with sample sizes and complexity comparable to the motivating RPPA study. Figure 2.2 shows estimated partitions under coagulation and refinement, arranged by cell line and drug (c, d). In a few cases, we get estimates that merge two (true) clusters together (e.g., in the estimated partitions $\boldsymbol{\delta}^1_{cd}$ and $\boldsymbol{\delta}^2_{cd}$ for $(c, d) = (1, 3)$ and for $(c, d) = (2, 1)$). In most cases, however, the underlying cluster structure is accurately recovered and we are able to correctly identify which proteins are affected by the respective drug (proteins corresponding to darkest shades of gray in plot (b)).

Figure 2.3 shows estimated cluster membership and mean responses (first two columns) in comparison with the simulation truth (last two columns) for one specific combination of cell line and drug $(c = 2, d = 3)$. Comparing the simulation truth in column 4 with the estimated means in column 2 we find a good fit for the data. With one less cluster in the refinement stage picked by BIC, the model merges the two new clusters (darkest shades of gray in columns 3 and 4) into only one (darkest shade of gray in columns 1 and 2), still providing a good fit with a more parsimonious model.



(a) Coagulation: $t \leq \tau^1_{cd\ell}$, $t > \tau^2_{cd\ell}$ (b) Refinement: $\tau^1_{cd\ell} < t \leq \tau^2_{cd\ell}$

Figure 2.2: Scenario 1: Simulation truth δ^u_{cd} (above horizontal lines) and posterior estimated $\overline{\delta}^u_{cd}$ (bellow horizontal lines). Each vertical bar corresponds to a gene, with colors (grayscale) representing their respective estimated cluster memberships.

Scenario 1a. We explore prior sensitivity with respect to the prior on the random partitions and the structure of partitions over time. We first repeat inference, still with the same data as in scenario 1; but with a different hyperprior on the random partition, namely $\boldsymbol{\pi}_1 \sim \text{Dir}(0.1, \ldots, 0.1)$ and $\boldsymbol{\pi}_2 \sim \text{Dir}(0.1, \ldots, 0.1)$. Comparing Figure 2.4(a) with the second column of Figure 2.3 we find no difference with respect to the estimated mean responses.

Scenario 1b. Alternatively we consider inference under a single random partition that remains invariable over time, that is, using $\boldsymbol{\delta}^2_{cd} = \boldsymbol{\delta}^1_{cd}$ for all c and d, but still allowing changing mean levels over time as in (2.2). Figure 2.4 summarizes the resulting inference by showing the simulated data and the estimated clusters, using the same format as first and fourth columns in

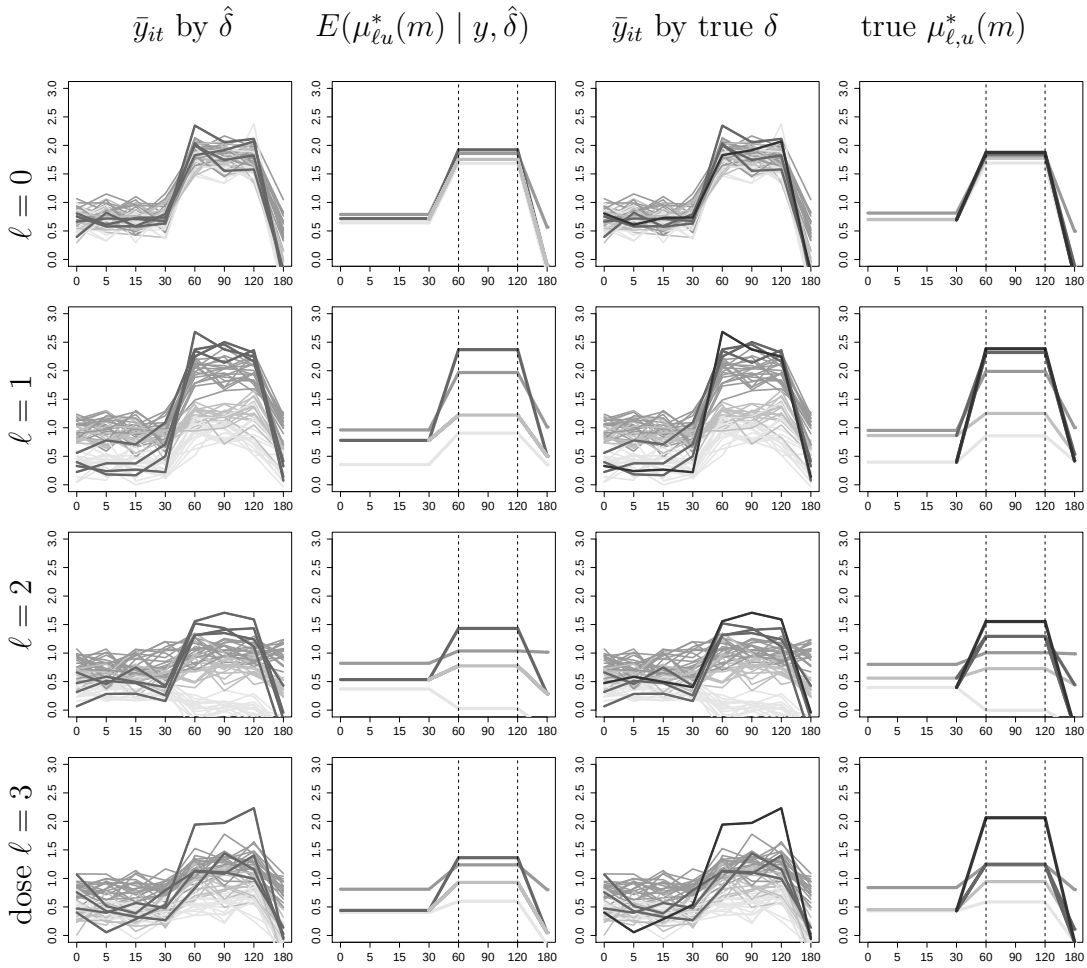
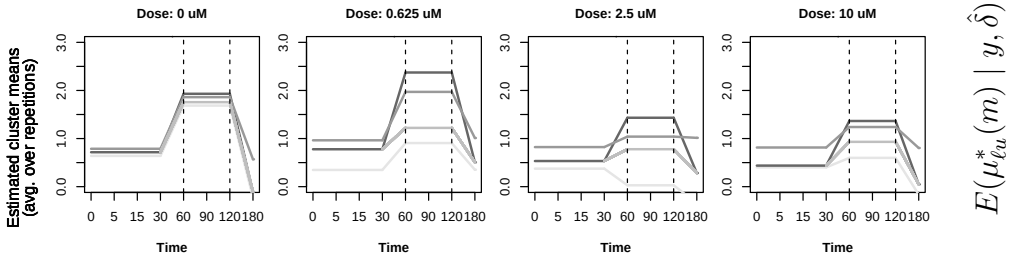
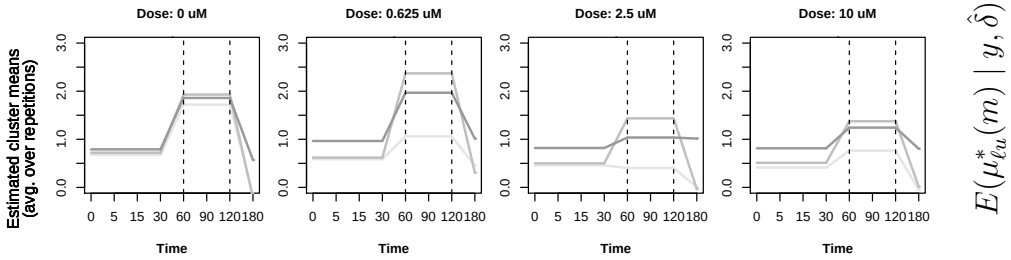


Figure 2.3: Scenario 1: Mean-response estimation for one of the simulated cases (cell line 2, drug 3). Each row corresponds to a different (increasing) dosage level. Time is measured in minutes on the horizontal axis. Column 1 and 3: average over repetitions, $\bar{y}_{it} = \frac{1}{J}\sum_{j=1}^{J} y_{\ell ijt}$ for the 55 proteins. Each line corresponds to a specific protein and the color (grayscale) indicates the posterior estimated cluster (column 1) and the true cluster (column 3). Column 2 and 4: posterior estimates (column 2) and simulation truth (column 4) for $\mu_{i\ell t}$ with dashed lines denoting estimated refinement and coagulation times τ^1_ℓ and τ^2_ℓ.



(a) $\boldsymbol{\pi}_1 \sim \text{Dir}(0.1, \ldots, 0.1)$ and $\boldsymbol{\pi}_2 \sim \text{Dir}(0.1, \ldots, 0.1)$.



(b) Single partition model: $\boldsymbol{\delta}^2_{cd} = \boldsymbol{\delta}^1_{cd}$.

Figure 2.4: Scenarios 1a and 1b: Inference under two variations of the prior model. Colors (grayscale) denote estimated clusters. Panel (a) shows inference under an alternative hyperprior with a symmetric Dirichlet prior, $\boldsymbol{\pi}_1 \sim \text{Dir}(0.1, \ldots, 0.1)$ and $\boldsymbol{\pi}_2 \sim \text{Dir}(0.1, \ldots, 0.1)$. Panel (b) shows inference using a single invariant partition of proteins over time, i.e., $\boldsymbol{\delta}^2_{cd} = \boldsymbol{\delta}^1_{cd}$.

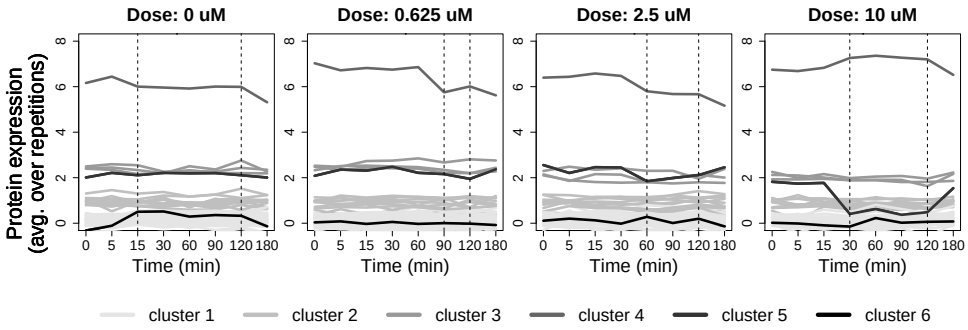
Figure 2.3. Comparing Figure 2.4(b) with the second column of Figure 2.3 shows a substantially deteriorated fit under the reduced model without the refined partition.

Scenario 2: We consider another hypothetical scenario with a simulation truth that closely mimicks the estimated effects in the actual RPPA data analysis. The data $y_{cdi\ell jt}$ are simulated with parameters fixed at the posterior estimates obtained in section 2.5, with $(\kappa_1, \kappa_2) = (4, 6)$. Figure 2.5 summarizes simulation results for $c = 2$ and $d = 1$. Using the BIC criterion we select $(\kappa_1, \kappa_2) = (4, 6)$, matching the simulation truth (with $(\kappa_1, \kappa_2) = (4, 7)$ and $(5, 6)$ being second and third best). Overall, the cluster-specific means are accurately estimated and the model fits the simulated data, indicating that inference under the proposed model can report meaningful summaries for the motivating RPPA data.

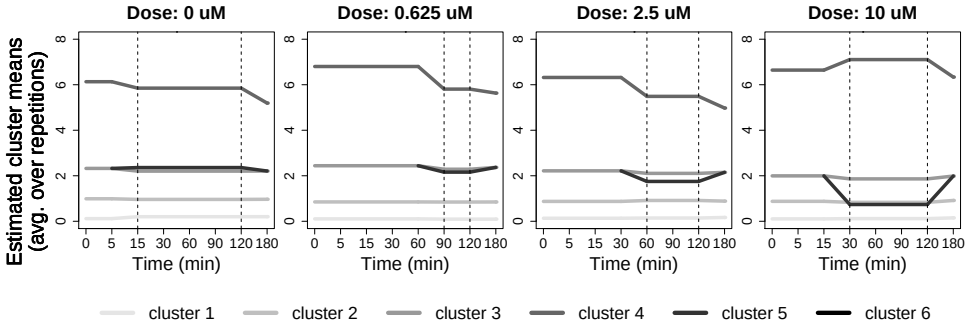
2.5 Proteomics Data

Based on BIC we select $(\kappa_1, \kappa_2) = (4, 6)$ (Figure 2.6). While more complex models (with more clusters) exhibit even better BIC, we find that the results for those models remain very similar to the ones obtained under $(4, 6)$, but with several empty and redundant clusters. We therefore proceed with the more parsimonious model.

We implement MCMC simulation for 5,000 iterations discarding the first 2,000 as initial burn-in. Hyperparameters are fixed as in the simulation



(a) Simulated data (over 4 doses). Data for each protein is shown as a connected line over time. Colors (grayscale) indicate cluster membership under the simulation truth.



(b) Estimated mean expression and cluster memberships. Colors (grayscale) indicate estimated cluster structure.

Figure 2.5: Scenario 2. Data (panel a) and estimated mean response and cluster membership (panel b).

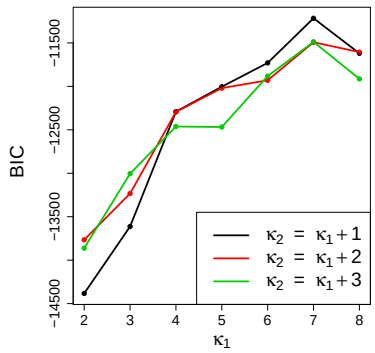


Figure 2.6: BIC for different number of clusters κ_1 and κ_2 (the bigger, the better).

study under Scenario 1, i.e., $\nu_\Sigma = 10$, $\boldsymbol{V}_\Sigma = I_{8\times 8}$, $a_\gamma = 1$, $b_\gamma = 1$, $\boldsymbol{\eta}_1 = (1, ..., 1)^\top \in \mathbb{R}^{\kappa_1}$, $\boldsymbol{\eta}_2 = (1, ..., 1)^\top \in \mathbb{R}^{\kappa_2 - \kappa_1}$, $\mu_{00} = 0$, $v_{00}^{-1} = 0.4444$, $a_v = 1$ and $b_v = 1$.

Figures 2.7 and 2.8 show estimates of the effect of PI3K inhibitor on the 55 proteins in cell lines 1 and 2 over time, respectively. Cell line 1 is the cell line MDA-MB-231 and cell line 2 is MDA-MB-468. Both are derived from a 51-year-old caucasian woman with metastatic breast cancer. The two cell lines have been shown to respond differently to chemotherapies and hormone therapies. Here, the goal is to characterize response to PI3K inhibition. The following discussion highlights related inference summaries. Keeping in mind the context of this analysis in the early phase of a drug development and the moderate sample sizes, inference should be understood as hypothesis generating, and findings should not be over-interpreted.

The first two columns in both figures are as in Figure 2.3 and show the

model fit to the data. Going from top to bottom (increasing dose) in Figure 2.7 one can see that protein S6 pS235/236 decreases with increased PI3Ki dose. At the same time HER2 is activated by the PI3K inhibitor. These two genes form singletons in our analysis. The inhibition of S6 and activation of HER2 after PI3K inhibition have been well reported in the literature (Podsypanina et al., 2001; Serra et al., 2011). Our analysis for cell line 1, MDA-MB-231 confirms these findings. In addition, we see that MAPK pT202Y204 is activated in this cell line as a result of PI3K inhibition. We see increased MAPK expression 5 minutes after the PI3K inhibitor is applied to the samples. The activation of MAPK as a result of PI3K inhibition is a major known discovery in breast cancer (Liu et al., 2009).

In contrast, results are different in cell line 2, MAD-MB-468 (Figure 2.8). MAPK is briefly inhibited by the PI3K inhibitor instead of being activated as in cell line 1. This suggests that cell line 2 includes a mechanism that might reverse the interactions of PI3K and MAPK. Due to large and complex down-stream pathways regulated by PI3K, the effects of its inhibition can be tissue-dependent and heterogeneous (Engelman, 2009). This is shown in the different response of protein expression in the two cell lines of this RPPA experiment. The differential response of MAPK to PI3K inhibition across the two cell lines could be important in interpreting the reason why they respond differently to therapies. Discoveries like this are expected to help biologists to set up new hypothesis for further testing.

Summarizing the refinement at time τ^1 as a distance between δ^1 and

δ^2 one could use, for example, the Hamming distance between co-clustering matrices P^1 and P^2 with entries $P^1_{ij} = I(\delta^1_i = \delta^1_j)$ and $P^2_{ij} = I(\delta^2_i = \delta^2_j)$ respectively. We find relative (to the number of pairs) Hamming distances between the partitions δ^1_{cd} and δ^2_{cd} to range from 0.01 to 0.03 depending on the cell line c and drug d.

Table 2.2 shows point estimates for the refinement and coagulation times τ^1_ℓ and τ^2_ℓ, respectively. For cell line 1 the three drugs (columns) behave similarly, causing the proteins to refine and revert to te baseline with a similar delay, without apparent dose effects. Cell line 2 is different from cell line 1 in that the cells in this line react heterogeneously to the three drugs and doses. In particular, cell line 1 seems to be more robust to the drugs as the refinement period is very short across doses. That is, the proteins in this cell line in general do not react to the drugs. For cell line 2, proteins seem to be more sensitive to the drugs. For the first three dose levels, 0, 0.625, and 2.5 μM, refinement starts earlier and ends later with increasing dose levels. This is expected as higher doses will lead to quick reaction and longer duration of the biological system. Dose level 10 μM is an outlier with a very short refinement period again. This might be due to the high potency of the high drug concentration (10 μM is the highest dose level).

Additionally, in Figure 2.9 we illustrate the benefit of the time-dependent clustering, with only two change points when the partition changes. In the figure we explore the use of independent clustering at each time point, using k-means for an easy implementation. While one could still identify a small

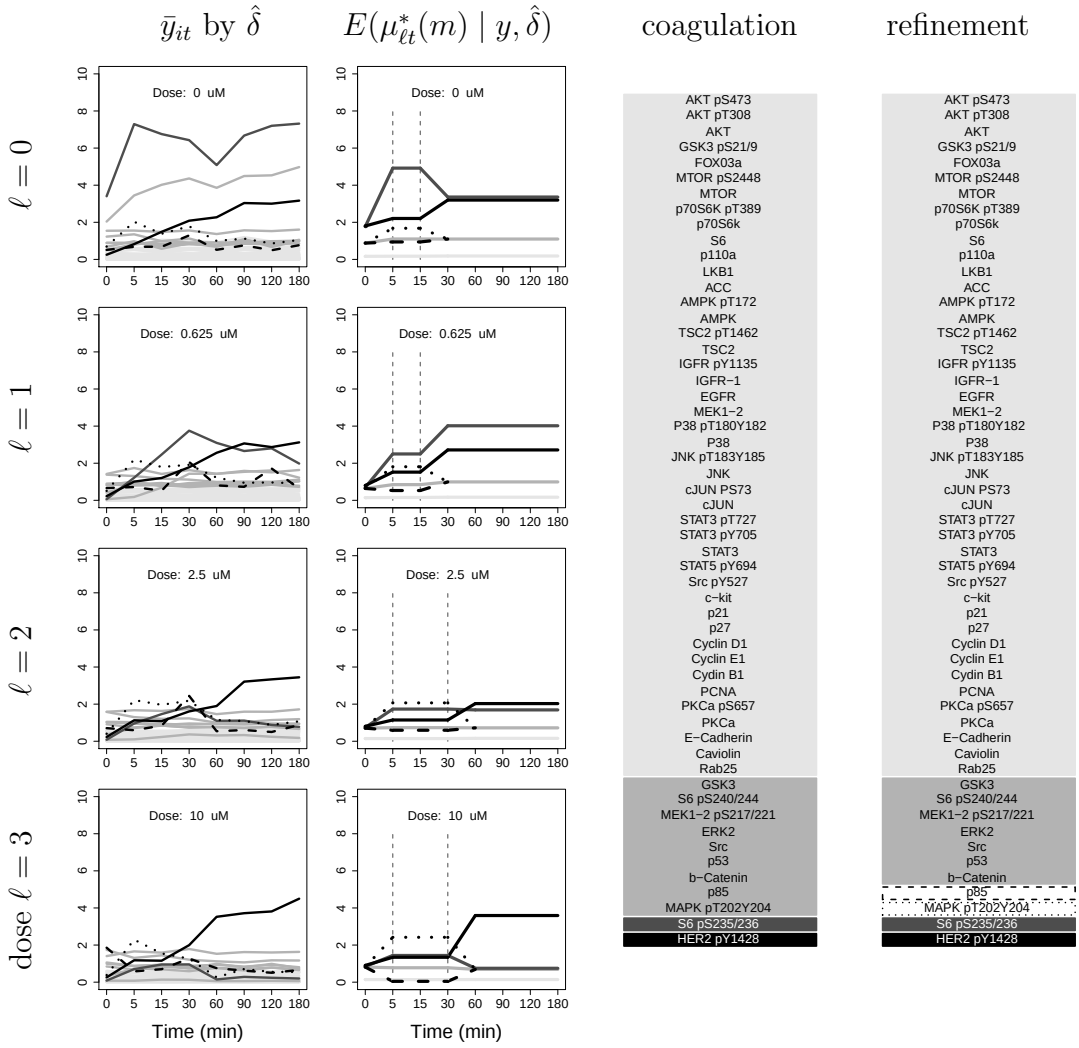


Figure 2.7: Results for proteins in cell line $c = 1$ exposed to PI3K inhibitor ($d = 1$). Colors (grayscale) denote distinct clusters with dashed lines corresponding to additional clusters formed at refinement. Columns 1 and 2 show $\bar{y}_{it}$ and $\mu_{i\ell t}$ as in Figure 2.3. The horizontal axis contains the observed times measured in minutes. Columns 3 and 4 show the original partition before refinement ($\boldsymbol{\delta}^1_{cd}$, column 3) and the refined partition after refinement ($\boldsymbol{\delta}^2_{cd}$, column 4).

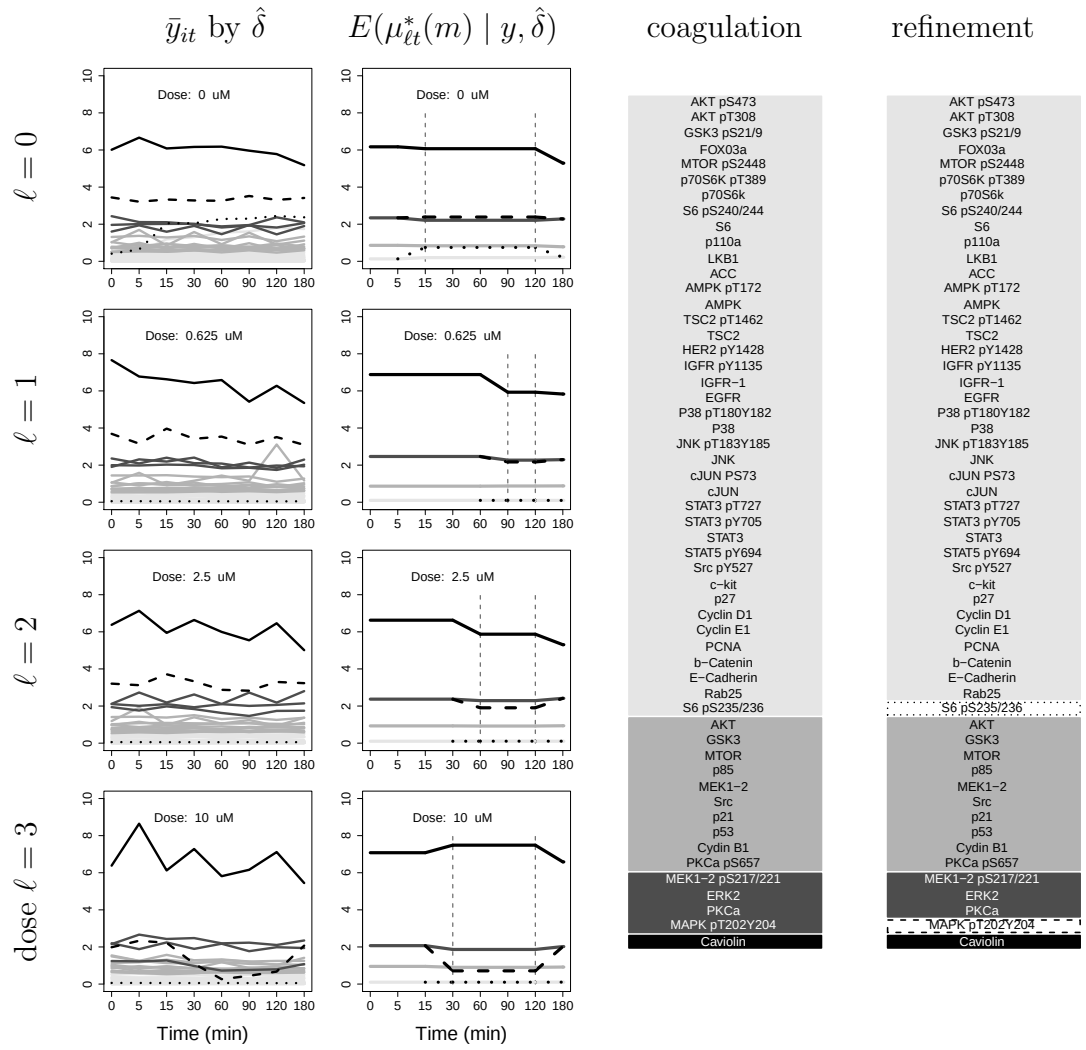


Figure 2.8: Same as Figure 2.7, now for cell line $c = 2$.

Table 2.2: Estimated (mode) refinement and coagulation times (minutes): τ_ℓ^u for cell line c, drug d, dose ℓ and $u \in \{1, 2\}$.

	cell line 1						cell line 2					
drug	PI3Ki		AKTi		MEKi		PI3Ki		AKTi		MEKi	
dose/time	refin.	coag.	refin.	coag.	refin.	coag.	refin.	coag.	refin.	coag.	refin.	coag.
0 uM	0	15	0	5	0	15	60	120	60	90	5	120
0.625 uM	0	15	0	5	0	5	60	90	5	90	60	120
2.5 uM	0	30	0	5	0	5	30	120	5	120	15	90
10 uM	0	30	5	30	0	5	0	30	90	120	30	60

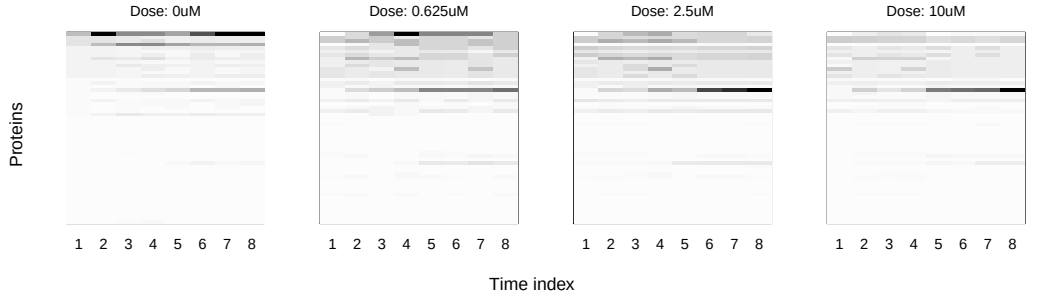


Figure 2.9: Independent k-means (k=5) estimation of partitions across time for different doses of PI3K inhibitor administered to cell line 1. For a fixed column (time index), each color represents the estimated cluster specific mean for that particular protein (higher expressions are darker).

number of proteins that seem to have their expression gradually increased or decreased at higher doses of the PI3K inhibitor, there is substantially more noise in the summary than in the corresponding plot in Figures 2.7 and 2.8.

2.6 Discussion

We introduced a model for finding a subpopulation of proteins that are most affected by a particular intervention. The key element of the model is a sequence of random partitions subject to the desired monotonicity. The same inference – without any change in the probability model – can be interpreted

as inference on mean protein expression over time, with the clusters serving the purpose of adaptively borrowing strength across proteins, doses, drugs and cell lines. The latter happens only at the level of hyperparameters. The approach is meaningful in any inference problem with a sequence of partitions that include a notion of monotonicity. It is most appropriate when limited data or high noise leaves more complexly structured models impractical to fit.

Several extensions and generalizations of the proposed model are possible. With more data one could replace the piecewise constant mean response by a piecewise linear mean response with little change in the remaining model. In the application to the RPPA data it was reasonable to assume that once the treatment effect wears off the mean response would revert to the initial levels. Other applications might call for lasting treatment effects, allowing for a different final level. Also, in other applications the use of more than two change points might be meaningful, with possibly different sequences of refinements and coagulations.

The main limitation is the computationally awkward problem of estimating the size of the partitions. With respect to the application, a limitation is that the described inference targets only mean expression levels, missing changes that are in the dependence structure. The latter is plausible when the intervention affects pathways and feedback mechanisms.

Chapter 3

Dependent Mixtures: Modelling Cell Lines

3.1 Introduction

We introduce a Bayesian mixture model with a dependent prior on the component-specific parameters. Most Bayesian inference for mixture models proceeds with independent priors on cluster-specific parameters that index the sampling model for each term of the mixture model. For a review of Bayesian inference in mixture models see, for example, Frühwirth-Schnatter (2006) and Frühwirth-Schnatter et al. (2019, Chapter 1).

There are some exceptions. Xu et al. (2016) argue for priors that favor diverse and parsimonious components in the mixture, which they implement using a determinantal point process. The idea is to favor mixture models with terms that define meaningfully different subpopulations. This becomes important if the inference aim is related to a biological interpretation of the underlying structure. While one can argue that asymptotically posterior inference in mixtures will concentrate on a parsimonious structure (Rousseau and Mengersen, 2011), this is not true for any finite sample size unless appropriate model assumptions are explicitly introduced in the model. In this paper we consider an inference problem that gives rise to a special type of parsimony in

a mixture model.

3.1.1 Modeling cell lineage data

The motivating application is inference for cell lineage data. The data comes from single-cell transcriptomics and allow inference on the evolution of cells as cell function and type evolve over the history of a cell. The finer resolution of single-cell assays (e.g., scRNAseq) in comparison with aggregated "bulk" data is crucial in applications that require cell-specific data, including examples of studies in immune systems (Stubbington et al., 2017; Miragaia et al., 2017), virus-host interactions (Cristinelli and Ciuffi, 2018), hematopoiesis (Wilson and Göttgens, 2018; Dharampuriya et al., 2017) among many others. In particular, single-cell assays such as scRNAseq allow to trace back the "history" of fully differentiated cells starting from their precursors, so they have become very important to the study of cell lineages (Stubbington et al., 2017).

A typical cell lineage dataset contains a sample of cells from a certain tissue along with cell-specific transcriptional profiles obtained, for example, from scRNAseq. Such profiles exhibit differences that are associated with the development stage of the cell. For instance, stem cells evolve into fully differentiated cells according to a process characterized by gradual transcriptional changes. Therefore, important differences are observed in transcription profiles along the path of development of the cell. Another potential application concerns temporal transformation of the cells, e.g. during cancer progression.

Lineage inference is then carried out to identify the underlying path of development from the initial state of the cell until its matured states. For a more detailed description of the objectives and challenges of lineage inference, see Korthauer et al. (2016) .

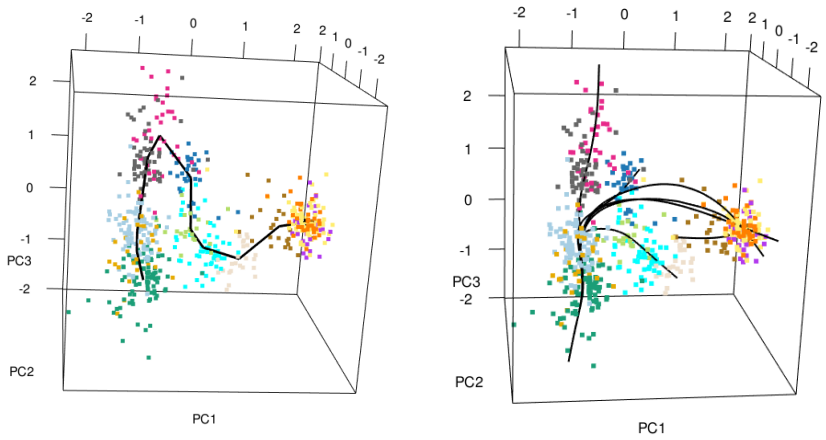


Figure 3.1: Left panel: Three-dimensional representation of single-cell gene expression profiles based on principal component analysis (data of Fletcher et al. (2017)); cells are colored by cluster. Right panel: results using the "Slingshot" method of Street et al. (2018).

Typically, the process of estimating the latent tree in cell lineage problems is done in 3 sequential steps. First, a dimension reduction method is applied to summarize information from the high dimensional scRNAseq data; then clustering of the cells is carried out in the reduced dimensional space; and finally, the latent tree is inferred given the clustering of the cells. Figure 3.1 illustrates a transformed data from a study of lineage.

Shiffman et al. (2018) introduce a generative model based on a Dirichlet diffusion process (Neal, 2003). This model, however, despite introducing a

notion of tree along which cell evolution occurs, does not restrict the trees to be simple and repulsive structures. Trees of a Dirichlet diffusion process are generated on a latent space via Brownian motion and thus do not enforce neither a partition nor a parsimonious representation of the data.

In contrast, Street et al. (2018) develop a method ("Slingshot") that takes as input a partition of the cell lines into different cell types and returns a smooth version of the underlying MST defined by the clusters' centroids: the paths from root to leaves are smoothed by principal curves. The Slingshot procedure requires observations to be clustered, which the authors suggest to do sequentially by clustering the cells with a k-means clustering, followed by the Slingshot construction of a minimum spanning tree on the cluster centroids. Such sequential approach (of first fixing the partition of cells, then using the centroids as nodes of the MST) is very common in the bioinformatics literature (REFERENCES?) however it assumes that the lineage structure (represented by the MST) does not play a role in clustering the cells, which is biologically not ideal.

In the following discussion we propose an inference approach that uses a model-based Bayesian perspective such as in Shiffman et al. (2018), but in the context of mixtures as in "Slingshot". That is, we propose a Bayesian dependent mixture model for cell lineage inference. The mixture components represent clusters of cells in the same development stage and the dependence is defined by a latent random tree with nodes that correspond to the centroid of the clusters. The cluster and lineage structures are modeled jointly and, as

a consequence, we allow the lineage to have an influence the clustering of cells. Full posterior inference on the clusters, the random tree and pseudotimes are obtained by Markov chain Monte Carlo (MCMC).

3.1.2 Dependent mixture models

The proposed model is a variation of Bayesian mixture models with dependent priors. Different clusters of cells should be dependent due to the fact that mixture components represent distinct intermediate states in the continuous process of cell development. Most literature on Bayesian mixture models assumes a priori independent cluster-specific parameters, with some exceptions. Xu et al. (2016) describe the application of determinantal point processes (DPP) (Kulesza and Taskar, 2012) in Bayesian mixture models as a way to impose repulsive stochastic behaviour for the prior on the mixture components. The motivation comes from the observation that similar mixture components do not make the model more flexible. Conversely, they only create redundancy and hurt the interpretability of the components. In the context of cell lineage data, the repulsive nature of the DPP would enforce the intermediate states of cell development to be dissimilar from each other.

Another motivation for dependent mixtures is the sharing of information between the different groups causing a shrinkage effect that is also a form of regularization. Mixed effect models is a key example of the use of hyperpriors for regularization (Lindstrom and Bates, 1990; Alston et al., 2012; Lachos et al., 2013). In Bayesian non-parametrics, hierarchical Dirichlet processes

(Teh et al., 2006) incorporate such shrinkage effect by assuming dependent component-specific Dirichlet processes (DP) $G_j \sim DP(\alpha_0, G_0)$ for mixture components $j = 1, \ldots, K$ with a common base measure G_0 that is itself a DP. Since DPs generate discrete random measures with the countably infinite support consisting of an iid sample from the base measure, it follows from the discreteness of G_0 that all $G_j,\ j = 1, \cdots, K$ will all share the same atoms.

An application of mixture models with predictor-dependent components is described in Chung and Dunson (2011). The authors define a Dirichlet process that assigns stick-breaking weights and atoms to random locations in predictor space, therefore obtaining random probability measures (which define the distribution of the mixture components) indexed by covariates in a continuous way.

Our proposal is to define dependence on the components of the mixture in a way that explicitly incorporates the biological structure that underlies cell lineage applications. We therefore propose the use of a random tree structure not only to explain the snapshot in the latent space of the continuous development of cells from its initial stage into mature differentiated cells, but also to model the dependence structure between the clusters of cells. Regularization is incorporated in the form a penalization on trees with too many nodes or with redundant edges. Our proposed model builds upon the slingshot model in Street et al. (2018) in which a Minimum Spanning Tree is calculated given the estimated centroids of the clusters of cells in a latent low dimensional space. The authors then use projections onto the MST to get a point estimate of the

pseudotimes for each cell. In contrast, by formally constructing a Bayesian mixture model with random trees on such latent space, we are able to provide full inference (with uncertainty captured by the posterior samples obtained through MCMC) on the clusters of cells, on the underlying tree structure and also on pseudotimes.

3.2 A Mixture Model for Inference on Cell Lineage

Let $\boldsymbol{Y}_i$ denote the recorded markers for the ith cell. In a study of cell lineage, the raw data could be biomarkers, i.e., protein levels for some selected proteins. The raw data are typically further processed by extracting, for example, the first few principal components which become the data $\boldsymbol{Y}_i$ in the upcoming discussion.

We start the construction of a Bayesian inference model by assuming a mixture sampling model for the $\boldsymbol{Y}_i$. Let $\boldsymbol{\theta}$ denote all unknown parameters. We assume

$$\boldsymbol{Y}_i \mid \boldsymbol{\theta} \overset{\text{iid}}{\sim} \sum_{j=0}^{k} w_j \; N(\boldsymbol{y}_i \mid \boldsymbol{\mu}_j, \Sigma_j). \tag{3.1}$$

The parameter vector includes, in particular, the number of terms in the mixture, $k+1$, the location parameters $\boldsymbol{\mu}_j$ and the covariance parameters for each cluster in the mixture model, $j = 0, \ldots, k$, and the relative weights w_j.

In words, we assume a mixture of normal sampling model for the data $\boldsymbol{Y}_i$, including cluster-specific covariance matrices Σ_j and cluster-specific location parameters $\boldsymbol{\mu}_j$. However, in the prior specification for the size of the

mixture k and for the cluster locations, the model construction parts ways with commonly used Bayesian inference for mixture models, which would continue with conditionally independent priors for the cluster-specific parameters. Instead we introduce a dependent prior across $\boldsymbol{\mu}_j$. Dependence is induced by the nature of the cell subpopulations being related as part of the cell differentiation.

Recall that the goal is to infer a structure that reflects the cell evolution path and its possible branching, starting with an original cell population indexed by $k = 0$ (and biologically known to be the root population). We represent this cell evolution path as a tree that includes the terms $\boldsymbol{\mu}_j$ in the mixture model (3.1) as the vertices, which are connected by edges that represent the cell differentiation. An additional set $\boldsymbol{b} = (b_1, \ldots, b_k)$ of indicator variables $b_j \in \{0, \ldots, k\}$ records the tree structure by specifying for each node the index of the parent node. The root node, $j = 0$ has no parent. A prior on the tree implicitely defines the prior probability model for the mixture component locations $\boldsymbol{\mu}_j$, by means of the following construction.

In order to define a meaningful notion of cell evolution, we need to carefully choose the form of such tree. This is achieved by defining a tree with a globally-dependent structure, which is able to induce repulsions between the branches and to avoid redundant components. For example, we avoid the possibility of a tree to grow back into itself. In fact, cell evolution follows a "monotonicity" requirement in the sense that cell characteristics evolve towards progressive degrees of differentiation, not going back to an undiffer-

entiated stage.

We introduce a preference for such structure using the notion of a minimum spanning tree (MST), whose origin traces back to Boruvka (1926). A MST is an edge-weighted, undirected graph that connects all vertices together, without any cycles and with the minimum possible total edge weight. The weight on an edge is the distance between the two nodes of the corresponding edge. In such a way, the most likely tree induces the desired parsimony requirement. In fact, given a set of nodes representing the different cell subpopulations, a MST can be seen as the most parsimonious way to represent the cell lineage.

We consider two alternative priors for $(k, \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k)$ based on a MST. One model is centered around trees that constitute a MST of the locations $\boldsymbol{\mu}_j$, but also allows trees that are not MST. The second model restricts the tree to be a MST, making $\boldsymbol{b}$ a deterministic function of $(\boldsymbol{\mu}_0, \ldots, \boldsymbol{\mu}_k)$.

3.2.1 Soft MST-dependent prior

Let ${}_k = (k, \mu_1, \ldots, \mu_k, b_1, \ldots, b_k)$ denote the tree, including cluster locations $\boldsymbol{\mu}_j$ (note that μ_0 is known and hence no prior is assigned to it) and tree structure $\boldsymbol{b}$. The soft-MST (s-MST) prior defines a dependent prior on the cluster locations as

$$p(\mu_1, \ldots, \mu_k, b, k) \propto \prod_{j=1}^{k} \{q(\mu_j)\} \exp\left\{-\alpha \sum_{j=1}^{k} d^2(\mu_j, \mu_{b_j})\right\} q(k). \tag{3.2}$$

In equation (3.2), $d(\mu_i, \mu_j)$ denotes the Euclidean distance between two nodes μ_i and μ_j. The terms $q(k)$ and $q(\boldsymbol{\mu}_j)$ are reference probability models. The penalty parameter α controls the level of shrinkage towards a MST, with $\alpha = 0$ implying no shrinkage and $\alpha \to \infty$ implying a deterministic restriction to MSTs.

It is important to notice that $q(k)$ and $q(\boldsymbol{\mu}_j)$ are not the marginal models for k and $\boldsymbol{\mu}_j$. The marginal distributions are implicitly determined by $q(k)$, $q(\boldsymbol{\mu}_j)$ and by the parameter α. In fact, conditionally on k, the model in (3.2) reduces to

$$p(\mu_1, \ldots, \mu_k, b|k) = \frac{1}{Z_k} \prod_{j=1}^{k} \{q(\mu_j)\} \exp\left\{-\alpha \sum_{j=1}^{k} d^2(\mu_j, \mu_{b_j})\right\},$$

where

$$Z_k = \int \cdots \int_{\mathbb{R}^{p\times k}} \prod_{j=1}^{k} \{q(\mu_j)\} \cdot \left[\sum_{b_1=0}^{k} \cdots \sum_{b_k=0}^{k} \exp\left\{-\alpha \sum_{j=1}^{k} d^2(\mu_j, \mu_{b_j})\right\}\right] d\mu_1 \ldots d\mu_k$$

is an intractable normalizing constant. The marginal prior on k can be written as

$$p(k) = \frac{Z_k \; q(k)}{\sum_{k=1}^{\infty} Z_k \; q(k)}.$$

It is easy to see that $p(\mu_1, \ldots, \mu_k, b, k)$ defined in (3.2) is proper (see Section C.1 in the Supplementary Materials).

The dependence among the centers is induced by the exponential term, that penalizes complex branching structures; for example, in trees with too many nodes or with redundant edges. Therefore, the prior can be seen as a regularization term, that is contraposed by the likelihood, which tends to

favour complex trees that provide better fit to the training data. The parameter α determines the strength of the regularization implied by the prior in comparison with the likelihood of the data. Implicit in (3.2) is the fact that the only branching structures b allowed are the ones that span trees with no internal cycles.

This joint prior, despite presenting an intractable normalizing constant, induces simple conditionals. For example, note that

$$p(b_j = i|\mu_1, \ldots, \mu_k, b^{(-j)}, k) \propto \exp\left\{-\alpha\ d^2(\mu_j, \mu_i)\right\}, \quad \forall i = 0, \ldots, k,\ \forall j = 1, \ldots, k \tag{3.3}$$

and that

$$p(\mu_j|\mu^{(-j)}, b, k) \propto q(\mu_j) \exp\left[-\alpha \left\{ d^2(\mu_j, \mu_{b_j}) + \sum_{l:b_l=j} d^2(\mu_l, \mu_j) \right\}\right]. \tag{3.4}$$

Equations (3.3) - (3.4) reflect the repulsive effect of the prior on the branching structure. In fact, the conditional distribution for b favours minimum spanning trees by assigning smaller probabilities to redundant structures, e.g. branches that grow back. This can be seen by the fact that each branch is selected to be the shortest (with larger probabilities) among those who preserve the spanning structure of the tree. In the case $\alpha \to +\infty$, this procedure has several analogies with Prim's algorithm (see Prim, 1957). In the opposite case, i.e. when $\alpha \to 0$, the prior on the trees is invariant with respect to the branching structure and the centers are independent. The model in this case corresponds to a finite mixture model with a prior on the number of components. In general, the model does a soft assignment of the branches to a MST structure (s-MST).

The conditional prior on the means, instead has a different effect. Each center is drawn from a linear combination of the independent prior term and the position of its parent and children. The larger the α parameter, the more evident the attraction towards the barycentre of parent and children. Moreover, note that if the distance d chosen is the squared euclidean, the conditional distribution of each μ_j is still normal, with updated parameters, i.e.

$$\mu_j|\mu^{(-j)}, b, k \sim \mathcal{N}\left[\frac{m_0/\sigma_0^2 + 2\alpha(\mu_{b_j} + \sum_{l:b_l=j}\mu_l)}{1/\sigma_0^2 + 2\alpha(1+f_j)}, \left\{\frac{1}{\sigma_0^2} + 2\alpha(1+f_j)\right\}^{-1}\mathcal{I}\right], \tag{3.5}$$

where f_j is the number of children of node j and (m_0, σ_0^2) are the prior hyperparameters of the group means. This is a fundamental feature that will imply posterior conditional conjugacy.

For the weights and the kernel covariance we use conditionally conjugate priors,

$$\begin{aligned} \Sigma_0, \ldots, \Sigma_k &\sim \text{IW}(\nu, \Psi) \\ (w_0, \ldots, w_k) \mid k &\sim \text{Dir}(\delta, \ldots, \delta) \\ \alpha &\sim \text{Exp}(\lambda_0) \\ k &\sim \text{Geom}(k-2|r_0),\ k \geq 2. \end{aligned} \tag{3.6}$$

3.2.2 Hard MST-dependent mixture model

The prior in (3.2) formalizes a preference for parsimonious structure by favoring mixture models with clusters that are connected by a tree with short cumulative length. By favoring shorter cumulative length the model shrinks

the tree structure towards a MST, but stops short of insisting on the tree actually being a MST. The model defines a joint prior $p(\boldsymbol{\mu}, \boldsymbol{b})$ on the cluster locations $\boldsymbol{\mu} = (\boldsymbol{\mu}_j)_{j=1}^k$ and the tree $\boldsymbol{b}$. An alternative model, named here as hard-MST (h-MST), defines a prior on $\boldsymbol{\mu}$ only by introducing the MST as a deterministic function of $\boldsymbol{\mu}$:

$$p(\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k \mid k) \propto \prod_{j=1}^{k} \left\{ N(\mu_j;\ \boldsymbol{m},\ \sigma_0^2 I) \right\} \exp\left[-\alpha \left\{ MST(\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k) \right\} \right]. \tag{3.7}$$

The term $MST(\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k)$ in equation (3.7) represents the minimum spanning tree with nodes $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k$ and edges $E_{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k} \subset \{1, \ldots, k\}^2$. The function denotes the total length of graph, which is defined by the sum of the squared lengths of its edges. Therefore, we have

$$(MST(\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k)) = \sum_{(j_1, j_2) \in E_{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k}} d^2(\boldsymbol{\mu}_{j1}, \boldsymbol{\mu}_{j2}).$$

By taking the lengths of the edges squared to define the length of the whole tree, we preserve conjugacy for the component specific means. By enforcing the MST structure the h-MST model provides stronger parsimony (in terms of favoring simpler tree structures) if compared with the s-MST. The parameter α regulates the strength of influence of the MST on the clustering structure: the higher its value,

Under the h-MST, the priors for the remaining parameters are the same

as described in section 3.2.1 equation (3.6) for the s-MST.

As mentioned in subsection 3.1.1, there is a crucial difference between our modeling approach in comparison with the two-step slingshot approach of Street et al. (2018) with regards to the dependence relationship of the cell lineage and the cluster centroids. Although the hard MST-dependent mixture model defines the MST as a deterministic function of the cluster centers, it implies a regularization effect of the tree structure on the distribution of the centroids by favoring cluster-specific means that lead to simpler (shorter) MSTs. On the other hand, the clustering step in Street et al. (2018) does not make use of the underlying lineage structure represented by the MST.

All full conditional distributions for this model are analytically available, except for $p(\boldsymbol{\mu}_j \mid \boldsymbol{y}, \text{rest})$ for which a Mtropolis-Hastings step was implemented. More details are available in Section C.3 in the Supplementary Materials.

3.3 Posterior Inference

In this section, we present the posterior simulation scheme to infer the tree structure, the optimal clustering configuration, as well as the estimation of the pseudotimes for each cell. For both the s-MST and h-MST models, the inference procedure is done via reversible jumps MCMC. In the next subsections we describe the inference procedure in more details under each model.

3.3.1 Inference under the s-MST Prior

Performing inference for a fixed size tree is straightforward under the soft-MST. In particular, the full conditionals are available in closed form and easy Gibbs sampling updates can be implemented (details are deferred to Section C.2 in the Supplementary Materials). For inference with an unknown number of components k we added a prior $k \sim \text{Geom}(k-2|r_0),\ k \geq 2$ and implement transdimensional posterior simulation, to accommodate the variable dimension of the parameter vector as k changes. The soft-MST prior uses reversible jump MCMC (RJ-MCMC) (Green, 1995) to implement transdimensional transitions. Proposing a tree also needs a proposal for the branching structure. However, given k nodes the total number of spanning trees is k^{k-2}, implying a curse of dimensionality for even fairly moderate values of k. This would in turn lead to inefficient proposals, which cause the algorithm to mix poorly.

In order to overcome this issue, we propose a variant of the RJ-MCMC which has been previously used in Xu et al. (2016) and Lee et al. (2015). Let us denote the parameters with $\theta_k = (\mu_1, \ldots, \mu_k, b_1, \ldots, b_k,$
$w_0, \ldots, w_k, \Sigma_0, \ldots, \Sigma_k)$. In the RJ-MCMC, a proposal that involves a change of k to $\tilde{k}$ would require to propose also a new set of parameters $\tilde{\theta}_{\tilde{k}}$. In practice, the joint proposal for a "new" dimension and a "new" set of parameters, decomposes in $q(\tilde{\theta}_{\tilde{k}}, \tilde{k}|\theta_k, k) = q(\tilde{\theta}_{\tilde{k}}|\tilde{k}, k, \theta_k)\ q(\tilde{k}|k)$. However, in many applications it is very hard to construct a suitable proposal, i.e. that gives significantly non-zero acceptance ratios, which in turn would lead to slowly mixing Markov

chains. This is mainly due to the curse of dimensionality that results from the space of possible branching structures b.

Given the ease of sampling from the posterior distribution of the parameters given a model dimension k, we can first perform MCMC runs with fixed dimensions. We then follow an idea from (Lee et al., 2015), who split the data into two parts: a small training set y' that serves the purpose of creating informative proposal distributions, and a test set y'' to evaluate the acceptance ratio. Let $p_1(\theta_k|y') = p(\theta_k|k, y')$ denote the posterior distribution under k using the training sample y'. We use p_1 in two instances. First, we replace the original prior term $p(\theta_k|k)$ and, second, we also use it as proposal distribution $q(\tilde{\theta}_{\tilde{k}}|\tilde{k})$. The test data y'' is then used to evaluate the acceptance probability. By the nature of the Metropolis-Hastings acceptance probability the proposal distribution and the prior factor in the target distribution cancel out, making this a feasible strategy, i.e.

$$\alpha = \min\left\{1; \frac{p(\tilde{k})\ p(y''|\tilde{\theta}_{\tilde{k}}, \tilde{k})\ q(k|\tilde{k})}{p(k)\ p(y''|\theta_k, k)\ q(\tilde{k}|k)}\right\}, \tag{3.8}$$

The strategy has an analogy with model comparison via fractional Bayes factors (O'Hagan, 1995).

3.3.2 Inference under the h-MST Prior

Inference under the h-MST with fixed tree size is similar to s-MST. Full conditionals are also analytically available, except for the component-specific means $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k$, which are sampled according to a Metropolis-Hastings step.

The idea is to propose a new value for component $\tilde{\boldsymbol{\mu}}_j$ from a proposal distribution $q(\tilde{\boldsymbol{\mu}}_j \mid \boldsymbol{y}, \boldsymbol{\mu}^{(-j)}$ that restricts to have the same MST determined by the current nodes $MT(\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k)$. The restriction of the branching structure to the MST also reduces the parameter dimension, which facilitates implementation of split-merge reversible jump to infer the unknown model dimension. The procedure is similar to Dellaportas and Papageorgiou (2006) and **?** in which the proposed split moves preserve first and second moment of a randomly chosen component $j \in \{1, \ldots, k\}$ of the mixture distribution that describe observations $\boldsymbol{Y}_i$. A few modifications in the prior on the group specific covariance matrices are required to apply Dellaportas and Papageorgiou (2006) and **?**. We define $\Sigma_j = \Lambda_j^\top$ to be the eigendecomposition of Σ_j i.e., the columns of are the eigenvctors and $\Lambda_j := \text{diag}(\lambda_{j1}, \ldots, \lambda_{jk})$ contains the respective eigenvalues. The set up requires the eigenvectors to be fixed and shared across all components. To handle such restriction, we fix to be the matrix of eigenvectors of the sample covariance matrix $\hat{\Sigma}$ of the data $\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_n$. The model specification is completed by assuming the following prior structure

$$\begin{aligned} \lambda_{jd}^{-1} &\sim \text{Gamma}(a_0, b_0) \\ (w_0, \ldots, w_k)|k &\sim \text{Dirichlet}(\delta, \ldots, \delta) \\ \alpha &\sim \text{Exp}(\lambda_0) \end{aligned} \tag{3.9}$$

3.3.3 Optimal partition

Once the MCMC chain approximating the posterior distribution is obtained, one of the goals is to attribute each observation to a cell subpopulation. Note that, although the posterior includes also values for the visited partitions $\{c^{(m)}\}_{m=1}^{M}$, finding a point estimate for a clustering configuration is non-trivial due to the cardinality of the space of partitions (Bell number). The posterior mode, for example, is not an adequate solution as each support point might have a negligible posterior probability. In the Bayesian literature, there exist many papers dealing with this problem. A common approach is to use a decision theoretic framework. In practice, one introduces a suitable loss function $L(c_n, \hat{c}_n)$ giving the cost of estimating the "true" c_n by $\hat{c}_n$. Then, a Bayes optimal estimate is given by any partition $\hat{c}_n$ which minimizes the posterior expectation of the loss function. In other terms, the loss is averaged across all possible true clusterings, where the loss associated to each potential true clustering is weighted by its posterior probability. Note that the posterior mode corresponds to the $0-1$ loss, i.e. $L_{0-1}(c, \hat{c}) = \mathbb{1}(c \neq \hat{c})$. This loss function is not satisfactory because a partition which differs from the truth in the allocation of only one observation is penalized the same as a partition which differs from the truth in the allocation of many observations. To alleviate this issue, Dahl (2006); Lau and Green (2007); Wade et al. (2018) propose different loss functions. In this work, we use the Variation of Information loss, whose theorerical results were developed in Wade et al. (2018).

3.3.4 Estimation of pseudotimes

Starting from a pre-specified root node (which in our case represents the cluster center of stem cell), pseudotime for a data point in the mixture is defined as the cumulative length of the shortest path starting at the root and ending at the closest projection of the data point onto the latent tree. Pseudotimes are a deterministic function of the latent tree structure. Since MCMC simulation produces a posterior sample of realizations of the latent random tree (as a function of locations $\boldsymbol{\mu}_j$ and branching structure $\boldsymbol{b}$), the same simulation output implies a posterior sample on pseudotimes. In the context of inference for cell lineage, inference on pseudotimes relates to the time a cell takes to develop from the initial state to the final differentiated state in distinct mature cell types.

3.4 Simulated Datasets

We here show results of inference for two simulated dataset under the proposed sMST and hMST models, as well as the slingshot method for comparison. The first dataset was generated as a mixture model from an underlying tree. The second one was instead used in Street et al. (2018) to demonstrate how accurate the recovered branching structure is.

3.4.1 Simulation 1

We first assess the model with a stylized example consisting of a dataset simulated via a mixture model on an underling tree (see Figure 3.2, left panel).

Soft MST

In Figure 3.2 (right) we show the posterior sampled trees, which seem to reconstruct well the underlying truth. In Figure 3.3 we show the marginal posterior on the number of groups (left), giving large probabilities to the true number of nodes ($K = 7$). Moreover, the density estimate in Figure 3.3 (right) is good.

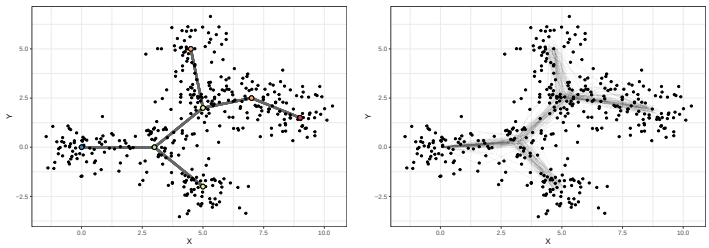


Figure 3.2: Fit of the s-MST model. The left panel shows the simulation truth. The right panel shows $M = 500$ posterior samples of τ_k.

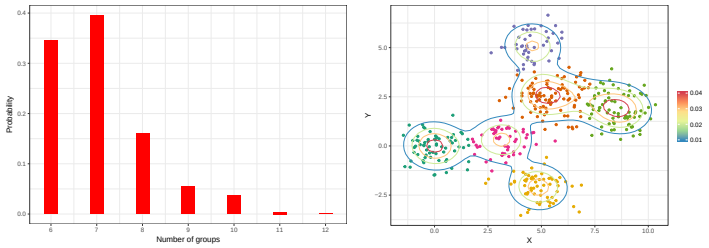


Figure 3.3: Left panel: posterior marginal distribution of the number of nodes (dimension of the model). Right panel: posterior density estimate obtained via the s-MST model. The observations are colored according to the optimal cluster labeling.

Hard MST

Inference under the hMST model is done via transdimensional MCMC according to sections 3.2.2 and 3.3.1 (full conditional distributions are described in appendix C.3). First, we run very short parallel MCMC chains (5 iterations) with fixed number of components K ranging from 2 to 15. In this first step, the full conditionals in appendix C.3 are computed using only the training data and transdimensional moves are not proposed. The second step consists of 10000 iterations of the transdimensional MCMC in which changes in the number of components K are proposed as in section 3.3.1 followed by the regular Gibbs sampling updates listed in the appendix C.3. Finally, cluster membership point estimates $\hat{\boldsymbol{c}} := (\hat{c}_1, \ldots, \hat{c}_n)$ are obtained based on those 10000 iterations following Wade et al. (2018). Fixing $\boldsymbol{c} = \hat{\boldsymbol{c}}$ (which also implies a fixed posterior estimate on K) we run the update steps 2-5 in C.3 for more 5000 iterations.

Table 3.2 shows the results of the estimation of K under different initial number of mixture components $K_0 \in \{2, 8, 15\}$ and different fractions of training data $\epsilon \in \{0.1, 0.25, 0.5, 0.75\}$ reserved for proposing transdimentional moves. We see that our approach is fairly robust, recovering the true number of clusters (seven) for most configurations of K_0 and ϵ.

Figure 3.4 shows the posterior estimates on the MST structure and cluster membership for $K_0 = 8$ and $\epsilon = 0.5$. The simulated true branching structure is well recovered by the hMST model. In contrast with the soft MST, for a small number of iterations, the hMST model exhibits an edge

Table 3.1: Estimated number of clusters K under the hMST model. The methodology of Wade et al. (2018) was applied to the first 10000 iterations of the transdimensional MCMC under different choices of ϵ (fraction of data reserved as training) and K_0 (value of K used in the initialization of the MCMC algorithm).

	$K_0 = 2$	$K_0 = 8$	$K_0 = 15$
$\epsilon = 0.1$	2	6	7
$\epsilon = 0.25$	3	7	7
$\epsilon = 0.5$	7	7	8
$\epsilon = 0.75$	7	7	6

between clusters 2 and 6, which happens as a consequence of enforcing the MST structure (in some occasions, 2 and 6 are closer than 4 and 6).

Table 3.2: Estimated number of clusters K under the hMST model. The methodology of Wade et al. (2018) was applied to the first 10000 iterations of the transdimensional MCMC under different choices of ϵ (fraction of data reserved as training) and K_0 (value of K used in the initialization of the MCMC algorithm).

	$K_0 = 2$	$K_0 = 8$	$K_0 = 15$
$\epsilon = 0.1$	3	6	6
$\epsilon = 0.25$	4	6	6
$\epsilon = 0.5$	6	6	6
$\epsilon = 0.75$	6	6	6

Slingshot

Figure **??** shows the results of applying k-means (k=7) for recovering the clustering of cells followed by the slingshot algorithm for inference on the

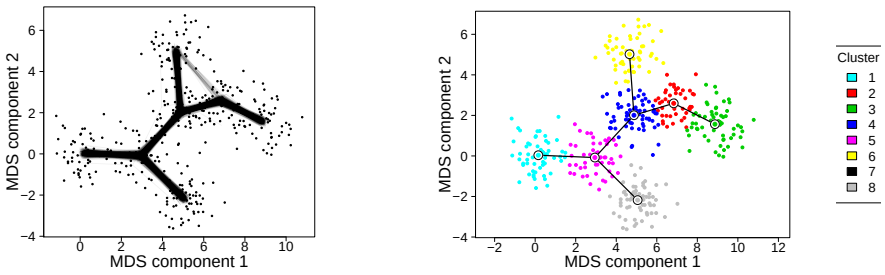


Figure 3.4: Estimated branching structure of the hMST model with $K_0 = 8$ and $\epsilon = 0.5$ based on the last 5000 MCMC iterations (i.e., cluster membership indicators fixed at posterior estimate). Left panel: stochastic tree estimates under the hMST model. Right panel: multiple runs of slingshot applied to the simulated data.

cell lineage. We can see that some initializations lead to cluster structures that do not correspond to the truth under simulation (see the first plot for example). This issue is fixed once we consider a higher number of random initializations and select the one with best value of the objective function (Figure **??**).

The slingshot algorithm is robust to the choice of K, as illustrated in Figure 3.5, specially when picking large values of K in the k-means algorithm.

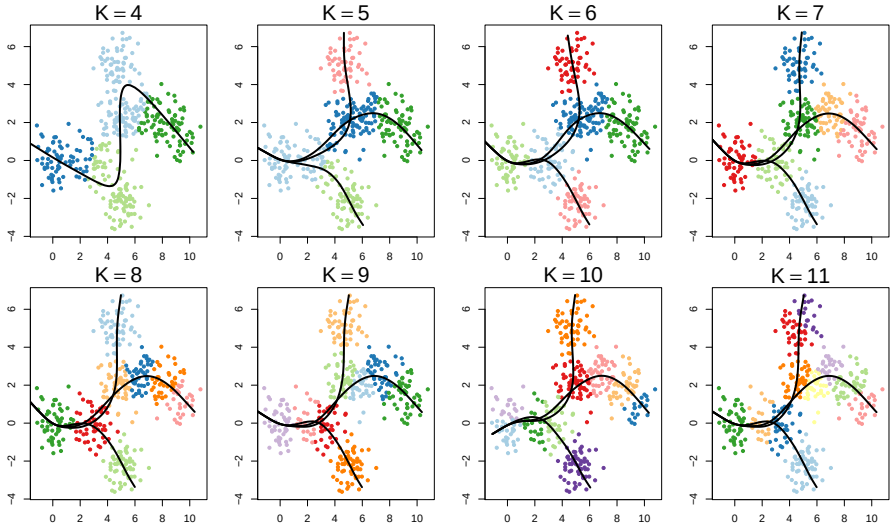


Figure 3.5: Parallel runs of slingshot applied to the simulated data for K ranging from 4 to 11. Clusters are estimated by the best result among 10 random initializations of the K-means algorithm.

3.4.2 Simulation 2

We now apply the algorithm to the same simulated dataset presented in Street et al. (2018). In Figure 3.6 we show samples from the posterior on the trees. From Figure 3.7 (left), representing the marginal posterior on the number of groups, one can see that the algorithm seems to be mixing well across the dimensions of the trees. Moreover, the density estimate in Figure 3.7 (right) is good.

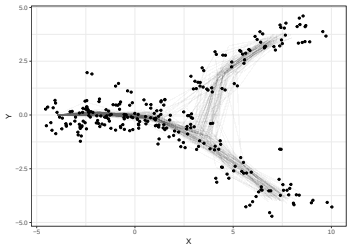


Figure 3.6: Plot of the posterior sampled trees.

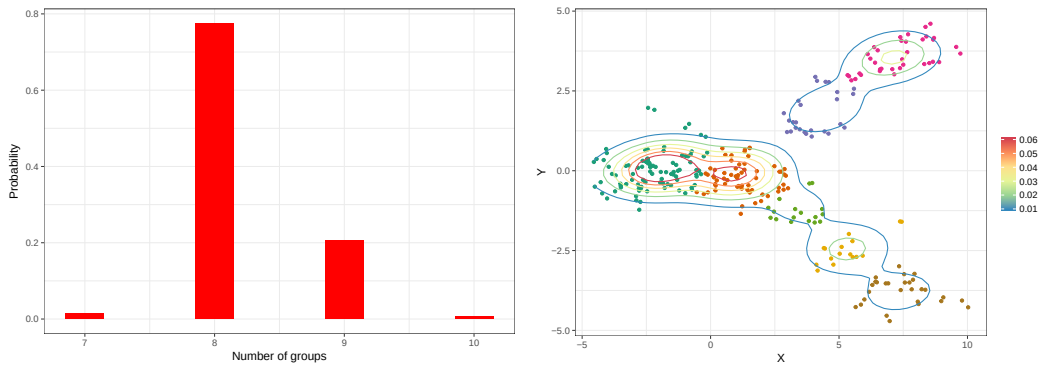


Figure 3.7: Left panel: posterior marginal distribution of the number of nodes (dimension of the model). Right panel: posterior density estimate obtained via the s-MST model. The observations are colored according to the optimal cluster labeling.

Hard MST

Again, we run 15000 iterations of MCMC on h-MST model. The first 10000 iterations include transdimensional proposal based on splitting the data into training and test, while the last 5000 are evaluated conditionally on the VI point estimate for the cluster membership structure. The h-MST model enforces more parsimony than the s-MST, which can be seen in Figure 3.8 as fewer components (five) are identified by MCMC.

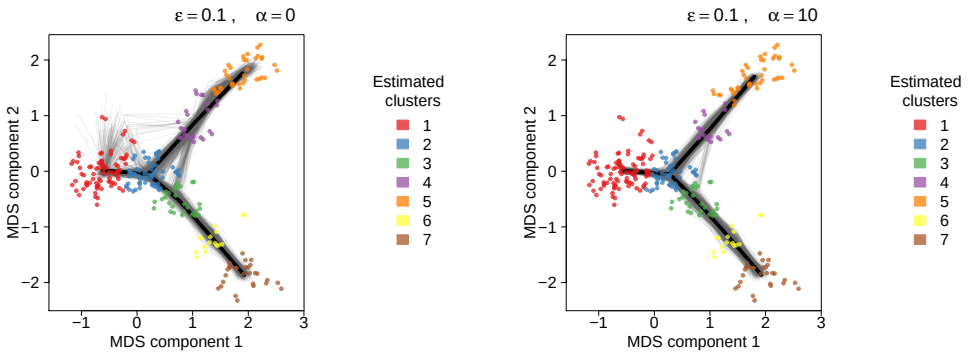


Figure 3.8: Results of posterior estimation of MST. Curves in gray are the posterior sampled MST and the black tree in the point estimate a posteriori. α represents the strength of regularization towards simple MST structures that is implied by the hMST prior on $\boldsymbol{\mu}$. ϵ is the fraction of the data reserved as training for the purpose of building the transdimensional proposals.

3.5 Mouse Data

We analyze data from a single cell RNA-seq experiment on horizontal basal cells (HBC) from the adult mouse olfactory epithelium (Street et al., 2018). The goal is to infer the continuous progression from stem cells into terminal mature cells and to estimate the cell-specific pseudotimes.

Due to the heterogeneity of cell populations, the analysis of traditional transcription data, such as bulk microarrays, does not allow researchers to discover cell dynamics. In fact, the underlying signal can be potentially masked when averaging over thousands of samples (Korthauer et al., 2016), possibly compromising statistical power.

The original data (before preprocessing) is available on GEO in GSE95601 and also in https://github.com/drisso/fletcher2017data. Preprocessing follows

the steps in (Perraudeau et al., 2017), which are listed here for completeness. The dataset originally contains measurements for 28284 genes throughout 849 cells. A total of 102 low-quality cells are removed from the dataset and the 1000 most variable genes are retained.

The resulting data is then normalized and the dimension is further reduced to 50 by fitting a Zero-Inflated Negative Binomial-based Wanted Variation Extraction (ZINB-WaVE) model following Risso et al. (2018). The ZINB-WaVE assumes a zero inflated negative bionomial model to extract low-dimensional signal from the data, accounting for dropouts (inflation of zeros), over-dispersion, and the count nature of the single cell RNA-seq data. Finally, multidimension scale (MDS) (Mardia et al., 1979) is applied to reduce the dimension further to 2 (this is the only deviation from Perraudeau et al. 2017 in which the dimension is reduced to 5). MDS consists of a rearrangement of the observations in a lower dimensional space (dimension 2 here) based on the matrix of pairwise distances computed using all the original 50 dimensions.

Hard-MST model. We start by showing results of application of the model that enforces MST structure (Section 3.2.2). The RJMCMC was run for 3000 iterations. The first 2000 are used to obtain a point estimate for the cluster membership indicators c_i according with Dahl (2006) and also for the number of mixture components K. The final 1000 iterations are run with fixed c_i and k. The hyperparameters were chosen as $r_0 = 0.5$, $a_0 = b_0 = 10$, $\sigma_0^2 = 1$, $\lambda_0 = 1$ and $\delta = 1$ to reflect non-informative prior knowledge.

Figure 3.9 shows the estimation of the underlying minimum spanning tree. We estimate 8 nodes with one branching leaving the main path of the spanning tree (green). The right pannel illustrates the posterior uncertainty on the edges of the tree and highlights the proximity of green cluster with both the purple and the yellow clusters.

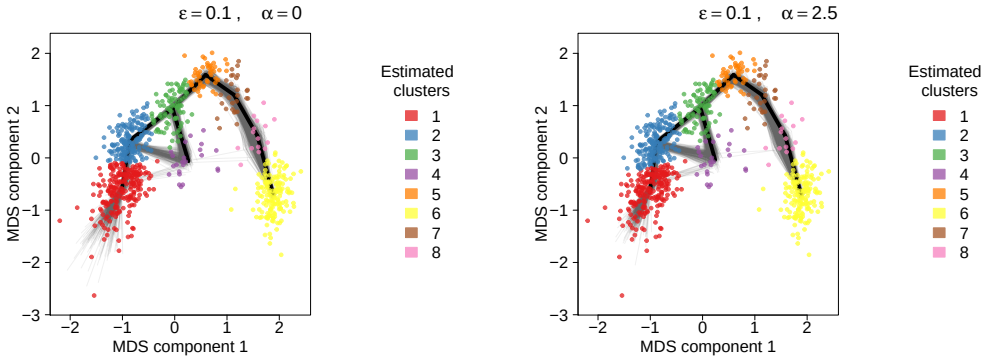


Figure 3.9: Posterior estimates of the latent MST and clustering membership structure based on the last 5000 MCMC iterations. Left panel: independent mixture ($\alpha = 0$). Right panel: MST dependent mixture ($\alpha = 2.5$).

We now focus on posterior estimation of pseudotimes. For each MCMC sample (after burn-in) we construct a posterior sample for the cell-specific pseudotimes as a deterministic transformation of the underlying MST. Such transformation is defined by calculating the distance along the tree from its root node to the projection of the cell onto the closest edge in the tree. Let $T_i(\tau)$ denote the pseudotime for cell i. Figure 3.10a illustrates the evaluation of $T_i(\tau)$, where the particular tree in the plot is fixed as the MST τ determined by the posterior point estimates for the cluster centers. The right panel shows marginal posterior standard deviations for the cell-specific pseudotimes,

conditional on cluster membership. Such graphical summaries help to identify those cells that are more prone to missclassification for being at approximately equal distance from two or more branches in the MST.

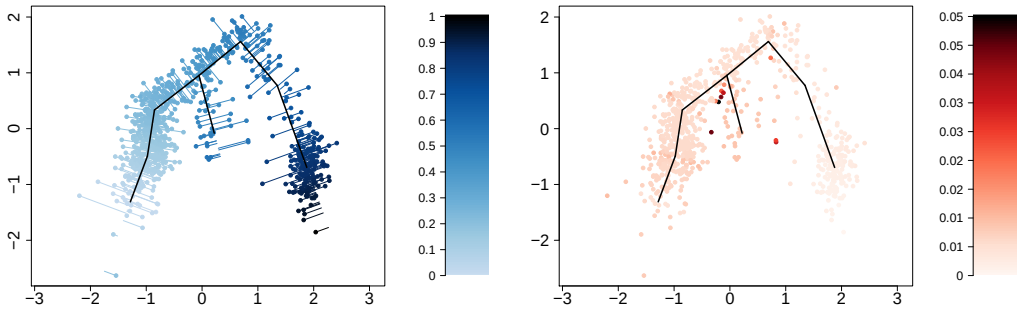


Figure 3.10: Left panel: Estimated pseudotimes for each cell. The extremes 0 and 1 were chosen arbitrarily. Right panel: posterior standard deviation of pseudotimes for each cell. Axis represent the two components of the MDS transformation.

In Figure 3.11, we can have a broader view of the estimated pseudotimes for cells in each one of the K clusters.

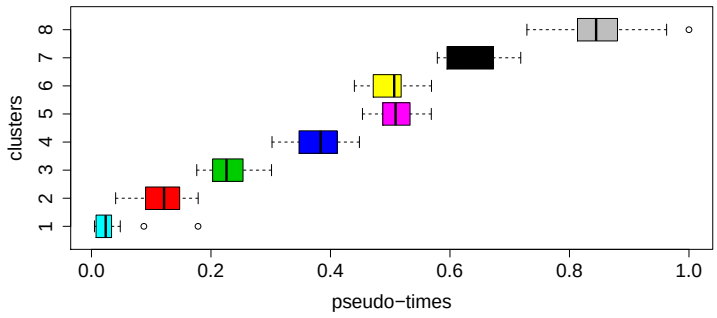


Figure 3.11: Cluster specific boxplots of median posterior pseudotimes obtained for each cell.

Slingshot. The slingshot method is a multistep algorithm that produces an underlying MST conditional on a fixed estimate of the cluster centroids. The algorithm first computes the MST with the clusters' centroids as nodes. Then it fits for each leaf a principal curve that smooths the path from the root to that leaf along the corresponding branches of the MST. Each principal curve represents a cell different development path.

We now investigate the sensitivity of the slingshot method to the clustering of cells. We apply multiple independent runs of k-means algorithm initialized at random with K=8. Figure 3.12 shows that the resulting MST is highly dependent on the initialization of the k-means algorithm, in some cases omitting important branches or creating artificial branches that clearly do not correspond to distinct cell populations. However, picking the best among multiple consistently solves the issue, as illustrated in Figure 3.13.

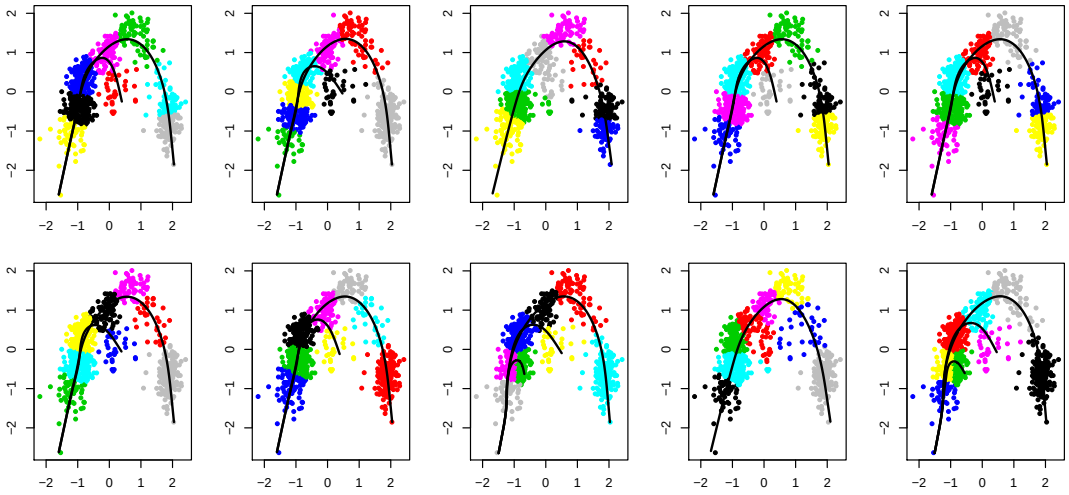


Figure 3.12: Multiple runs of slingshot applied to the mouse data. Each plot corresponds to a distinct random initialization of k-means algorithm (K=8). Axis represent the 2 MDS components for dimension reduction.

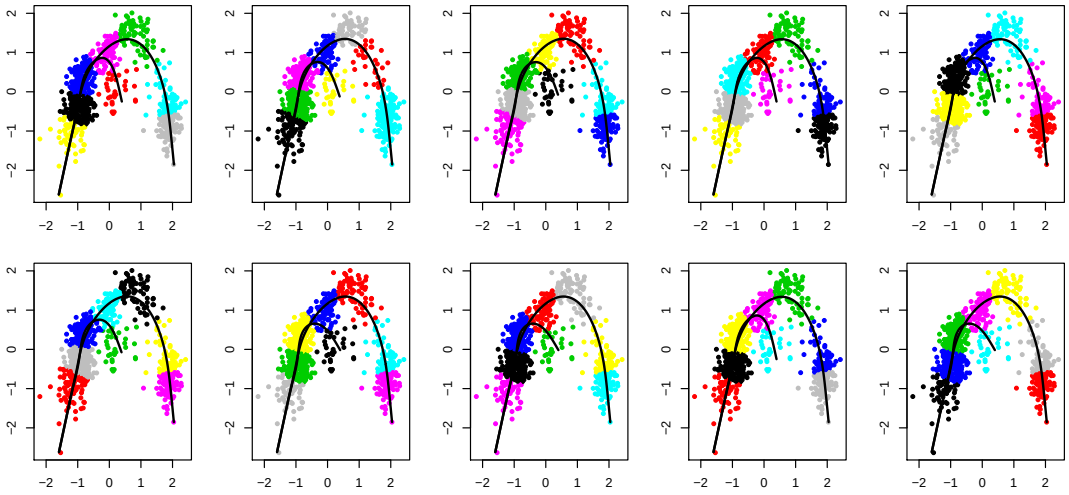


Figure 3.13: Multiple runs of slingshot applied to the mouse data. Each plot corresponds to the best result among 10 distinct random initializations of k-means algorithm (K=8). Axis represent the 2 MDS components for dimension reduction.

3.6 Discussion

We developed a dependent mixture model for single cell RNA sequencing data to estimate cell lineages. The proposed model takes into account the underlying tree in the transformed data on a lower dimension when defining the cluster structure of the cells: the model penalizes cluster allocations that define over complex trees. We presented two forms of defining such penalization terms: under soft MST or hard MST.

Chapter 4

A Two Step Bayesian Model for Matching Cell Line and Patient Genomic Profiles

4.1 Introduction

In a precision medicine paradigm, the patient's specific genetic architecture is assessed to propose a personalized treatment that is expected to be optimal for that individual. In this context, there is a trend in modern medicine to move from generalized treatment approaches towards the tailored treatment strategy that is dictated by their genomics or molecular profile. This paradigm shift accelerates the needs for advances in pharmacogenomics technology and associated analytical methods. In this paper, we develop methods to meet this demand, inclduing in particular novel priors for random structures. Briefly, we propose developing an integrative statistical framework, that merge multiplatform genomics ('omics, in short) profiles from multiple model systems (e.g. patients and cell-lines) for finding significant drug targets, preclinical models for appropriate drug discovery and repurposing and, finally, to calibrate therapeutic potential for future patients. By identifying these similarities (and differences) across model systems, we are able to gather more refined information about the patient than what is contained in their specific profile, while still proposing a personalized treatment that is strongly tied to

the patient's profile - through appropriate integration of various data sources.

The objective of this paper is then to construct a novel Bayesian statistical approach for matching patient gene profiles with cell line profiles. Such inference is needed, among many other applications, for data integration, precision medicine and patient specific treatment assignment. The expansion of modern medicine and fast growth of research in health sciences have led to a great increase in available data on multiple sources/platforms such as The Cancer Genome Atlas (TCGA, `tcga-data.nci.nih.gov`), Cancer Cell Line Encyclopedia (CCLE, `portals.broadinstitute.org/ccle`), International Cancer Genome Consortium (ICGC, `icgc.org`) to name a few. A model-based approach for matching of a patient profile with data from other sources, such as from cell lines, allows us to access a wider range of information to predict a patients response to specific treatments. Important for the envisioned application, the matching should be carried out on the basis of a biologically meaningful signal only, putting aside mere noise.

A cell line is a culture of cells extracted from a tissue (e.g., cancer cells from a tumor in a human tissue) and grown in an in-vitro environment that simulates the environment of the tissue in the organism where it was extracted. Therefore, cell lines serve as models to study cancer biology. Information from the response to a drug or treatment applied to cell lines (cultivated *in-vitro*) is used to infer about the expected response *in vivo* (Goodspeed et al., 2016). Similarly, individuals can be grouped according to similarities between their profiles and observed profiles in a fixed set of cell lines. In such scenarios, the

mapping of cell lines and patients opens the possibility to construct treatment recommendations based on results for the corresponding cell lines (Sinha et al., 2015). We propose a statistical approach that seamlessly combines the output of the Bayesian mixture model based on a proposal by Parmigiani et al. (2002) with a novel two-way Bayesian non-parametric (BNP) mixture model that is constructed as an extension of a BNP bi-clustering model of Lee et al. (2013).

Parmigiani et al. (2002) propose a Gaussian-uniform mixture model for probability of expression (POE). Later in the model construction we shall use the latent trinary signal of the POE model to carry out nested clustering of patient samples and the desired matching with cell lines. The uniform component in the POE models havier tails associated with genes that are over- and under-expressed, while the Gaussian term corresponds to regularly expressed genes. The authors argue that, by trichotomizing gene expressions into these 3 categorical levels, the POE approach smoothly removes uninformative biological and instrumental noise that is naturally present in genomic profile data, therefore strengthening downstream analysis.

The clustering of patient samples and the desired match with cell lines builds on a model developed of Lee et al. (2013), who present a Bayesian model (NoB-LoC) that identifies genes (columns) that are relevant for clustering of samples/individuals (rows). The identified genes are then partitioned in such a way that genes within the same subgroup (column-wise clusters) give rise to a common nested partition of individuals (row-wise clusters). The approach is motivated by the observation that high-dimensional protein profiles make it

hard to find meaningful clusters of samples/individuals. Researchers therefore often restrict attention to groups of proteins that are expected to lead to more meaningful and interpretable results. NoB-LoC identies such groups in a seamless process, together with the nested clustering of samples. The NoB-LoC model conveniently allows for different clustering of samples with respect to different groups of proteins. In our context, this translates to association of cell lines and patients depending on the set of proteins in the profile.

Developing the outlined model consturction, this chapter makes two major contributions: the first one is the integration of POE with the two-way clustering building on the NoB-LoC model. The second, and perhaps more important contribution is the extension of the NoB-LoC model to allow for explicit probabilistic matching of profiles that could come from distinct sources (e.g., cell lines and patients). In short, in the proposed approach we first use the NoB-LoC model to partition the proteins according to a zero enriched Pólyia urn process where some proteins are set aside as inactive proteins, while the selected proteins are grouped into protein clusters (active proteins). Within each protein cluster, the samples are partitioned again, using a partition model that matches patients to cell lines. The motivation is that the usually high-dimension protein profiles make it hard to find similar samples to be clustered together, therefore restricting the attention to groups of proteins is expected to lead to more meaningful and interpretable results. This procedure also naturally allows for identification of co-expressed proteins in the form of protein clusters, i.e. a group of genes that are biologically correlated

are also expected to have their expression levels "tied together" along different samples. Finally, the NoB-LoC model conveniently allows for different clustering of samples depending on the subsample of proteins that is considered. In our problem, this translates to association of cell lines and patients depending on the set of proteins in the profile.

The real data used in the statistical analysis comes from an experiment using reverse phase protein arrays (RPPA) which record the expression of selected proteins simultaneously on multiple cell lines and patients samples (Charboneau et al., 2002). The dataset analyzed here consists of lung cancer protein expressions (233 proteins) measured in 687 patients and 124 cell lines. Data is batch corrected, i.e., they are also adjusted for the batch effect difference between cell line and patients' data).

4.2 POE Model

In this section we describe the POE (probability of expression) model defined in Parmigiani et al. (2002). We modified some of the priors in order to obtain analytical full-conditionals for as many parameters as possible, which facilitates the MCMC implementation in the larger, encompassing model (more details ahead and also in appendix D.1).

Each observation y_{sg} consists of expression levels for protein (gene) $g \in \{1, \ldots, G\}$ and sample $s \in \{1, \ldots, S\}$. Latent variables e_{sg} indicate high expression of gene g in sample t ($e_{sg} = 1$), normal expression ($e_{sg} = 0$) and under expression ($e_{sg} = -1$). Each possible value of e_{sg} determines a differ-

ent distribution for the observed gene expressions according to the following Gaussian-Uniform mixture model

$$(y_{sg} \mid e_{sg}) \sim \begin{cases} Unif(\alpha_s + \mu_g,\ \alpha_s + \mu_g + k_g^+), & \text{if } e_{sg} = 1, \\ N(\alpha_s + \mu_g,\ \sigma_g^2), & \text{if } e_{sg} = 0, \\ Unif(\alpha_s + \mu_g - k_g^-,\ \alpha_s + \mu_g), & \text{if } e_{sg} = -1. \end{cases}$$

The lengths k_g^+ and k_g^- of the support of the uniform components should cover the tails of the corresponding gene expression distribution implying heavier tails than the Gaussian distribution. Under normality, the great majority of the samples (probability 0.997) concentrate within 3 standard deviations from the mean; therefore the constraints $k_g^+ > k_0\sigma_g,\ k_g^- > k_0\sigma_g$ imply heavier than Gaussian tails for fixed values of k_0 greater than, say, 3.

We now define the weights for each term in the mixture by the probability vectors $\boldsymbol{\pi}_g := (\pi_g^-, \pi_g^0, \pi_g^+),\ g \in \{1, \ldots, G\}$ where $\pi_g^+ = P(e_{sg} = 1 \mid \boldsymbol{\pi}_g)$, $\pi_g^0 = P(e_{sg} = 0 \mid \boldsymbol{\pi}_g)$ and $\pi_g^- = P(e_{sg} = -1 \mid \boldsymbol{\pi}_g)$. We assume $(\boldsymbol{\pi}_g \mid \boldsymbol{\eta}_\pi) \sim Dirichlet(\boldsymbol{\eta}_\pi)$.

Figure 4.1 illustrates the implied mixture model in the context of density estimation. The augmentation of the parameter space with inclusion of indicatior variables e_{st} allows for identification of up- and down-regulated genes that are not well captured bythe light tails of a single Gaussian component.

Posterior probabilities of differential expression are determined by Bayes

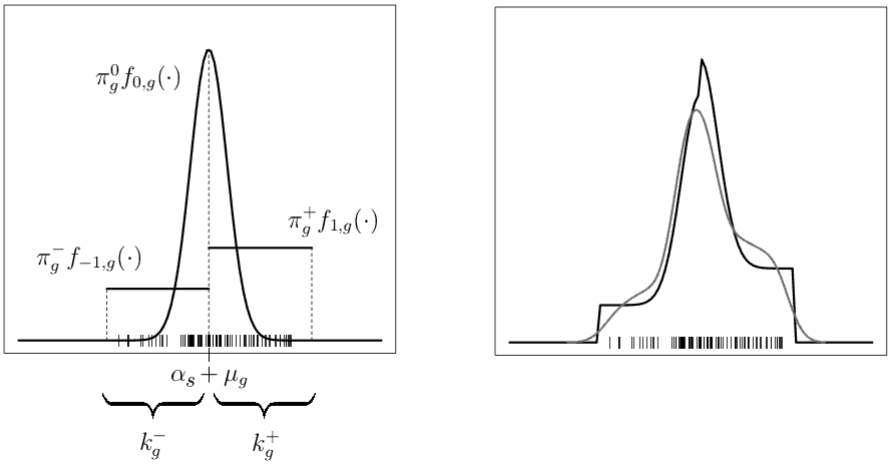


Figure 4.1: Left panel: Weighted components of the Gaussian-Uniform mixture model. Right panel: density estimates using Gaussian-Uniform mixture (black line) and using kernel estimate (gray line). Vertical bars represent data generated from the Gaussian-Uniform mixture.

rule as

$$\begin{aligned} p_{sg}^+ &:= P(e_{sg} = 1 \mid y_{sg}, \boldsymbol{\pi}_g, f_{1,g}, f_{0,g}) \\ &= \frac{\pi_g^+ f_{1,g}(y_{sg})}{\pi_g^+ f_{1,g}(y_{sg}) + \pi_g^0 f_{0,g}(y_{sg}) + \pi_g^- f_{1,g}(y_{sg})} \times \mathbb{1}(y_{sg} \in S_{f_{1,g}}) \\ &= \frac{\pi_g^+ f_{1,g}(y_{sg})}{\pi_g^+ f_{1,g}(y_{sg}) + \pi_g^0 f_{0,g}(y_{sg})} \times \mathbb{1}(y_{sg} \in S_{f_{1,g}}), \end{aligned} \tag{4.1}$$

where $\mathbb{1}(\cdot)$ is the indicator function and $S_{f_{1,g}}$ denotes the support of $f_{1,g}$. Analogously,

$$\begin{aligned} p_{sg}^- &:= P(e_{sg} = -1 \mid y_{sg}, \boldsymbol{\pi}_g, f_{-1,g}, f_{0,g}) \\ &= \frac{\pi_g^- f_{-1,g}(y_{sg})}{\pi_g^- f_{-1,g}(y_{sg}) + \pi_g^0 f_{0,g}(y_{sg})} \times \mathbb{1}(y_{sg} \in S_{f_{-1,g}}). \end{aligned}$$

Equations (4.1) and (**??**) are used for visualization of sample specific gene profiles. Since p^+_{sg} and p^-_{sg} are not simultaneously positive, the differences $d_{sg} := p^+_{sg} - p^-_{sg}$ will fall in the interval $[-1, 1]$, therefore serving as a unidimensional measure of gene expression ($d_{sg} \approx 1$ for highly expressed and $d_{sg} \approx -1$ for underexpressed genes).

and also for our sequential inference approach to serve as input for the followup application of the NoBloC model.

The model is completed by the prior specification $(\mu_g \mid \theta_\mu, \tau_\mu) \sim N(\theta_\mu, \tau_\mu)$, $(\alpha_s \mid \mu_\alpha, \tau_\alpha) \sim N(\mu_\alpha, \tau_\alpha)$ restricted to $\sum_{s=1}^{S} \alpha_s = 0$, $(\sigma^2_g \mid \gamma, \lambda) \sim InvGamma(\gamma, \lambda)$, $(k^+_g \mid \alpha_{k^+}, \beta_{k^+}) \sim InvGamma(\alpha_{k^+}, \beta_{k^+})$, $(k^-_g \mid \alpha_{k^-}, \beta_{k^-}) \sim InvGamma(\alpha_{k^-}, \beta_{k^-})$. We also chose prior models for hyperparameters as $\theta_\mu \sim N(m_\mu, s^2_\mu)$, $\tau_\mu \sim InvGamma(a_{\tau_\mu}, b_{\tau_\mu})$, $\alpha_{k^+} \sim Exp(\lambda_{\alpha_{k^+}})$, $\alpha_{k^-} \sim Exp(\lambda_{\alpha_{k^-}})$, $\beta_{k^+} \sim Gamma(a_{\beta_{k^+}}, b_{\beta_{k^+}})$, $\beta_{k^-} \sim Gamma(a_{\beta_{k^-}}, b_{\beta_{k^-}})$.

The motivation to propose Inverse Gamma priors for k^+_g, k^-_g and Dirichlet prior for $\boldsymbol{\pi}_g$ is to make use of conjugacy results in the full-conditional posterior of these parameters, which was not originally explored in Parmigiani et al. (2002).

4.2.1 Posterior inference for the POE model

We implement posterior inference by MCMC simulation. All full conditionals are available in closed form due to the choice of contditionally conjugate priors/hyperpriors; the only exceptions are α_{k^+} and α_{k^-}. We therefore implement Gibbs sampling transition probabilities for all parameters except

$(\alpha_{k^+} \mid \boldsymbol{y}, else)$ and $(\alpha_{k^-} \mid \boldsymbol{y}, else)$. For the latter we use Metropolis-Hastings transition probabilities with random walk proposal on $\log \alpha_{k^+}$ and $\log \alpha_{k^+}$ respectively. See appendix D.1 for more details.

To avoid numerical instability when sampling from truncated inverse gamma distributions (full conditional posterior distributions for k_g^+ and k_g^-), we used a variable augmentation scheme proposed in Damien and Walker (2001). The prior on the auxiliary variables introduced by the authors imply full-conditional posterior distributions for those variables, which are sampled together with the original parameters of the POE model within the full MCMC algorithm.

4.3 Nonparametric Bayesian Clustering with Patient and Cell Line Matching

4.3.1 A nested random partition and matching structure

Following posterior simulation for the POE model, the posterior estimated values d_{sg} become the inputs for model-based clustering of proteins and samples and the desired pairing with cell lines. This is implemented by the construction of a nested partition model that builds on the Nonparametric Bayesian local clustering (NoB-LoC) model defined in Lee et al. (2013). In this section, we relabel the data d_{sg} as d_{ig}^c if sample s corresponds to the i-th cell line or as d_{ig}^p if it is the i-th patient. The subindex g still denotes protein g. We will assume the dataset contains G proteins and S samples, including N^p patient samples and N^c cell line samples ($N^p + N^c = S$).

The model first partitions the proteins according to a zero enriched Pólya urn. One special cluster (corresponding to the zero-enrichment) is interpreted as "inactive proteins". The remaining ones are grouped into protein clusters (active proteins). Within each of these protein clusters, the samples are partitioned again by a second, nested partition model. The nested partition model includes also the desired pairing of each patient sample cluster with a matching cell line.

Consider the cluster membership indicator w_g for each protein $g = 1, \dots G$ and denote $\boldsymbol{w} = (w_1, \dots, w_G)$ the vector of protein cluster indicators. Let π_0 be the probability of inactivation and let $\alpha_0 > 0$ be the potential for creating a new cluster. Finally, define $n_k := \#\{g;\ w_g = k\}$ the number of proteins that fall into protein cluster k, for $k = 0, 1, \dots, K_{\boldsymbol{w}}$ with $k = 0$ denoting the cluster of inactive proteins and $K_{\boldsymbol{w}}$ denoting the total number of active clusters of proteins determined by $\boldsymbol{w}$. Then the zero enriched Pólyia urn defines $p(\boldsymbol{w} \mid \pi_0)$ as

$$p(\boldsymbol{w} \mid \pi_0) = \pi_0^{n_0}(1-\pi_0)^{G-n_0} \times \frac{\alpha_0^{K_{\boldsymbol{w}}} \prod_{k=1}^{K_{\boldsymbol{w}}} \Gamma(n_k)}{\prod_{g=1}^{G}(\alpha_0 + g - 1)}, \tag{4.2}$$

and for short we write $(\boldsymbol{w} \mid \alpha_0, \pi_0) \sim ZEPU(\alpha_0, \pi_0)$.

For each cluster of proteins defined by $\boldsymbol{w}$, two dependent partitions of the samples are defined, the first one involving patients only, and the second one (which is stochastically dependent on the partition of patients) includes only cell lines. The cluster membership indicators for the i-th patient and i-th

cell line in k-th cluster of proteins are defined as $\delta_i^{p,k}$ and $\delta_i^{c,k}$, respectively. The two partitions are then determined by the cluster membership indicators for patients and for cell lines. We marginally model the cluster membership of patients $\boldsymbol{\delta}^{p,k} := (\delta_i^{p,k})_{i=1}^{N^p}$ as $\boldsymbol{\delta}^{p,k} \sim ZEPU(\alpha_{pk}, \pi_{pk})$. This implies a random number J_k of active clusters of patients within the k-th group of proteins.

Conditionally on $\boldsymbol{\delta}^{p,k}$, several choices of priors on $\boldsymbol{\delta}^{c,k}$ are possible, each of them representing one way of matching cell lines' to patients' profiles.

Discrete uniform prior: we assume a discrete uniform prior for $\boldsymbol{\delta}^{c,k}$ on the patient samples and the set of inactive samples: $\delta_i^{c,k} \mid \boldsymbol{\delta}^{p,k} \sim Uniform(\{0, 1, \ldots, J_k\})$.

Discrete uniform prior with at most ℓ cell lines per cluster of samples: the conditional prior on $(\boldsymbol{\delta}^{c,k} \mid \boldsymbol{\delta}^{p,k})$ has the p.m.f

$$p(\boldsymbol{\delta}^{c,k} \mid \boldsymbol{\delta}^{p,k}) \propto \mathbb{1}(\boldsymbol{\delta}^{c,k} \in \mathcal{B}_\ell^{c,k}) \tag{4.3}$$

with support

$$\mathcal{B}_\ell^{c,k} = \left\{ (\delta_i^{c,k})_{i=1}^{N^c} \in \{0, 1, \ldots, N^c\}^{N_c}; \quad \textstyle\sum_{i=1}^{N^c} \mathbb{1}(\delta_i^{c,k} = j) \leq \ell \ \forall j \in \{1, \ldots J_k\} \right\}.$$

In the special case $\ell = 1$ for example, the multiplicative normalization constant in equation (4.3) is $\sum_{n=0}^{\min\{J_k, N^c\}} \binom{J_k}{n} \binom{N^c}{n} n!$.

Conditional zero inflated Polya urn: a conditional zero inflated Polya urn prior distribution is assumed prior for $\boldsymbol{\delta}^{c,k} \mid \boldsymbol{\delta}^{p,k}$ as a continuation of the process of clustering patient samples. This means that the initial probability

of allocation of cell lines to a patient cluster is proportional to the cluster size. Stochastically, this approach is equivalent to a joint zero inflated Polya urn for all samples.

We consider the Gaussian sampling model $(d^c_{ig} \mid \theta^c_{ig}, \sigma^2_g) \sim N(\theta^c_{ig}, \sigma^2_g)$ and $(d^p_{jg} \mid \theta^p_{ig}, \sigma^2_g) \sim N(\theta^p_{jg}, \sigma^2_g)$ for cell line i and patient j and protein g. The prior for θ^x_{ig} with x being either c or p is

$$\theta^x_{ig} \sim \begin{cases} I_{\theta^*_{jg}}, & \text{if } \delta_i^{x,w_g} = j > 0 \text{ and } w_g > 0 \text{ (active prot. and smpl.)} \\ N(\mu_{1g}, \sigma^2_{1g}), & \text{if } \delta_i^{x,w_g} = 0 \text{ and } w_g > 0 \text{ (active prot. inactive smpl.)} \\ N(\mu_{2g}, \sigma^2_{2g}), & \text{if } w_g = 0 \text{ (inactive prot.),} \end{cases} \tag{4.4}$$

where I_x denotes the point mass distribution (Dirac measure) at x. In (4.4) (first equation) we define the unique mean responses θ^*_{jg} for active proteins and samples. We denote by J_k the number of active sample clusters for all proteins g such that $w_g = k$. Notice that active cell lines and patients share the same mean response if they belong to the same sample cluster.

For the purpose of deriving the MCMC algorithm for posterior inference, we marginalize the patient specific (and cell line specific) means θ^x_{ig} in (4.4), which implies the following data distribution

$$d^x_{ig} \sim \begin{cases} N(\theta^*_{jg}, \sigma^2_g) & \text{if } \delta_i^{x,w_g} = j > 0 \text{ and } w_g > 0 \text{ (active prot. and smpl.)} \\ N(\mu_{1g}, \sigma^2_{1g} + \sigma^2_g), & \text{if } \delta_i^{x,w_g} = 0 \text{ and } w_g > 0 \text{ (active prot. inactive smpl.)} \\ N(\mu_{2g}, \sigma^2_{2g} + \sigma^2_g), & \text{if } w_g = 0 \text{ (inactive prot.).} \end{cases} \tag{4.5}$$

The prior for the unique mean response values is specified as $\theta^*_{jg} \sim N(\mu_{0g}, \sigma^2_{0g})$ with hyperpriors $\sigma_g^{-2} \sim Gamma(a_g, b_g)$, $\tau_{lg}^{-2} \sim Gamma(a_{lg}, b_{lg})$ and $\mu_{0g},\ \mu_{1g},\ \mu_{2g} \overset{iid}{\sim} N(m_0, s_0^2)$.

4.3.2 Summarizing the posterior nested partition

Point estimates of the cluster-membership indicators are obtained using the approach proposed by Dahl (2006). We run the MCMC algorithm and, after judging (practical) convergence, we evaluate for each pair $i < j$ of tumors within gene g, the pairwise co-clustering probability $\hat{p}_{ij} = \frac{1}{K}\sum_k p_{ijk}$, where K is the Monte Carlo sample size and p_{ijk} is an indicator for i and j being allocated to the same cluster, i.e., $p_{ijk} = 1 \Leftrightarrow e_{gi} = e_{gj}$ during iteration k. The dependence on g is omitted from the notation for clarity. The p_{ijk} and the $\hat{p}_{ij}$ are combined into $(I \times I)$ matrices $\boldsymbol{P}^{(k)} = [p_{ijk}]$ and $\hat{\boldsymbol{P}} = [\hat{p}_{ij}]$. We then report as posterior estimated $\bar{\delta}$ the partition corresponding to the co-clustering matrix $\boldsymbol{P}^{(k^*)}$ that minimizes $||\hat{\boldsymbol{P}} - \boldsymbol{P}^{(k)}||$. In other words,

$$\boldsymbol{P}^{(k^*)} = \arg\min_k ||\hat{\boldsymbol{P}} - \boldsymbol{P}^{(k)}||.$$

That is, k^* indexes the Monte Carlo sample whose co-clustering matrix is closest to $\hat{\boldsymbol{P}}$. The procedure for choosing k^* is done independently over each gene g.

4.4 Simulation

4.4.1 Simulation 1: POE

We carry out a first simulation to validate inference under the POE model.

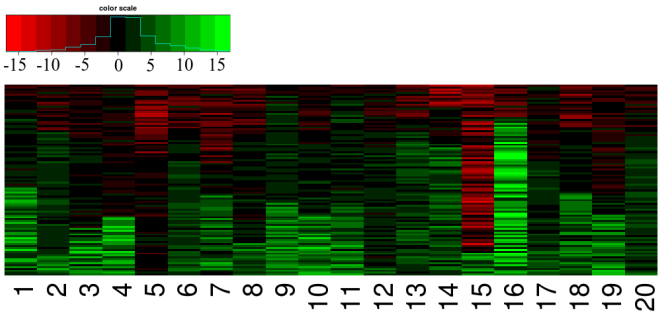
We simulate a dataset with 100 samples and 20 genes assuming the POE model as the underlying truth. Some small changes were done in the simulation process that deviates slightly from the model in section 4.2. Namely, σ_g^2 was sampled from $\sigma_g^2 = N(0, 0.25)^2 + 1$ instead of an Inverse Gamma prior and k_g^+, k_g^- were both sampled from $\max(Gamma(8,1),\ 5\sigma_g)$ instead of $\max(InvGamma(8,1),\ 5\sigma_g)$. Hyperparameters were fixed as $\boldsymbol{\eta}_g = (1,1,1)$, $\mu_\alpha = 0$, $\tau_\alpha = 0.5$, $\theta_\mu = \tau_\mu = 1$.

To carry out the MCMC inference procedure, we fix $\eta_g = (1,1,1)$, $\mu_\alpha = 0$, $\tau_\alpha = 100$, $a_{\tau_\mu} = b_{\tau_\mu} = a_{\beta_{k^+}} = b_{\beta_{k^+}} = a_{\beta_{k^-}} = b_{\beta_{k^-}} = \lambda_{\alpha_{k^+}} = \lambda_{\alpha_{k^-}} = 0.01$, $\gamma = \lambda = 0.1$, $m_\mu = 0$ and $s_\mu^2 = 100$. Such values were chosen to represent weak prior information.

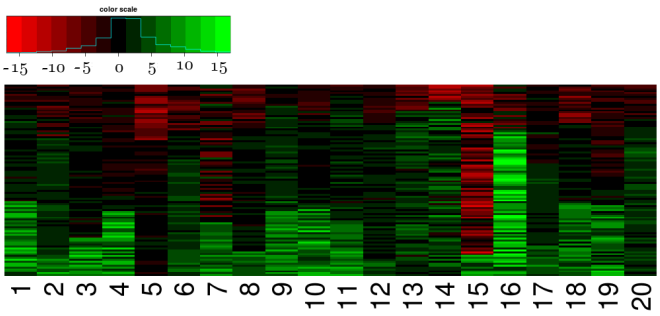
Figure 4.2 shows that the estimated cluster membership assignment of the observations reasonably recovers the simulation truth (compare pannels (a) and (b)). Panel (c) shows how the POE model removes noise from the data and highlights the biologically meaningful levels of protein activation (low, medium, high).

Figure 4.3 shows the density estimates a posteriori for 4 genes, comparing the true protein-wise cluster assignment with the point estimates obtained

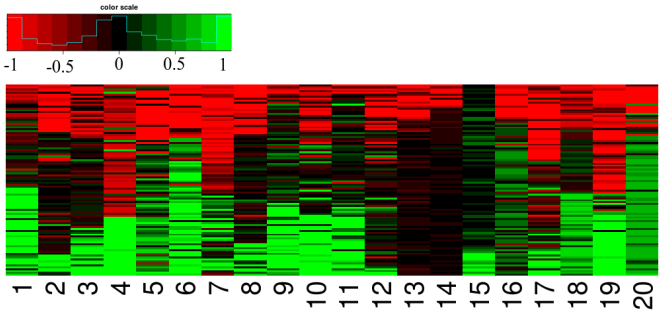
with the methodology from Dahl (2006) as described also in 4.3.2. The cluster membership indicators are typically well recovered. Notice that when 2 components are enough to estimate the underlying density among the different samples, we might have the absence of one of the groups of proteins with low, medium or high expression (see $g = 15$ for example). Also, the use of uniform components can, in some cases, exhibit high density near the center of the resulting mixed distribution therefore producing samples of medium expression even when the true is $e_{sg} = \pm 1$ in the simulation.



(a) Simulated y_{sg} ordered by true e_{sg}.



(b) Simulated y_{sg} ordered by estimated e_{sg}.



(c) d_{sg} ordered by true e_{sg}.

Figure 4.2: (a): Simulated data y_{sg}. Samples in each column are sorted by true e_{sg}. (b) Simulated data y_{sg}. Samples in each column are sorted by estimated $E(e_{sg} \mid \boldsymbol{y})$. (c) Differences $d_{sg} = p^{+}_{sg} - p^{-}_{sg}$ with the same ordering as in panel (a). The ordering of samples change according to the protein (column) but is the same throughout the 3 panels.

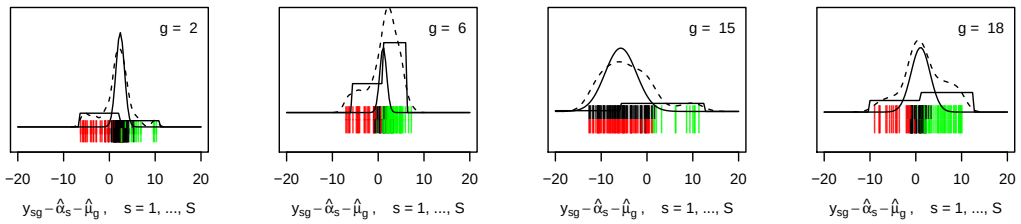


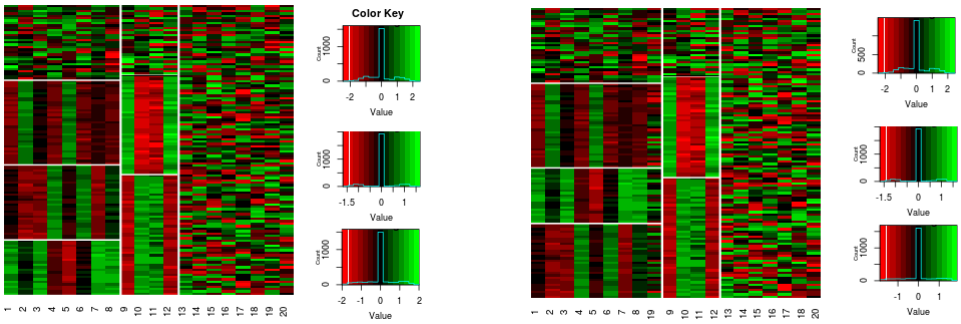
Figure 4.3: Posterior density estimates. Vertical bars represent centralized gene expressions $y_{sg} - \mu_g - \alpha_s,\ s = 1, ..., S$ for all genes g collored according to its estimated cluster membersip indicators (top) and true cluster membership indicators (bottom). Full lines represent the best fitting uniform and normal components of the mixture a posteriori multiplied by the respective weights. Dashed line corresponds to a kernel density estimate based on the vertical bars. Color code: black = -1, red = 0, green = 1.

4.4.2 Simulation 2: nested partitions

In this section we describe the simulation to validate inference on the NobLoc model. We replicated the scenario in Lee et al. (2013), with 100 samples and 20 proteins. The simulation truth incorporates the local clustering feature of first partitioning proteins and then within protein cluster, partitioning the samples. However, instead of simulating the protein and sample partitions according to a zero inflated Plya urn, we fixed the partition of proteins upfront to have two active protein clusters, the first one containing proteins with 3 active sample clusters; and the second containing 4 proteins with 2 active sample clusters. The cluster-membership assignment of samples was made uniformly at random among the available sample clusters. The cluster specific means were fixed at the same values in Table 1 of Lee et al. (2013). Inactive samples and proteins were all sampled from $Unif(-0.8, 0.8)$.

The standard deviation of the Gaussian sampling model for active samples was fixed at $\sigma_g = 0.1$ For an illustration, see Figure 4.4 panel (a).

In Figure 4.4 panel (b) we can see that the underlying cluster structure was reasonably captured by the NobLoc model. The only discrepancy is the inclusion of protein 19 in the first active cluster together with proteins 1 - 8, instead of classifying it as an inactive protein according to the simulation truth.



(a) Simulated y_{sg} ordered by true cluster assignments.

(b) Simulated y_{sg} ordered by estimated cluster assignments.

Figure 4.4: (a): Observations ordered according to the simulation truth. (b): Observations ordered according to estimated cluster membership indicators a posteriori. In both panels, rows represent samples while columns represent proteins.

4.5 Lung Cancer Dataset

4.5.1 The data

The dataset consists of protein profiles coming from an RPPA experiment on lung cancer samples. The data records 233 proteins that were preselected for their biological relevance to the study of this type of cancer. The

data records samples from 687 patients and 124 cell lines. The objective is to identify groups of similar patients and cell lines with respect to subgroups of co-expressed proteins (we informaly say that proteins are co-expressed if their expressions are correlated). We therefore expect that the samples (patients and cell lines) can be partitioned in a different way depending on the group of co-expressed proteins that is considered.

4.5.2 Results

We describe here the results of a joint inference of the POE model and nested clustering by NoB-LoC. We start by analyzing the results of directly applying the NoB-LC model to the original lung data (without running POE first), which is illustrated in Figure 4.5 (a).

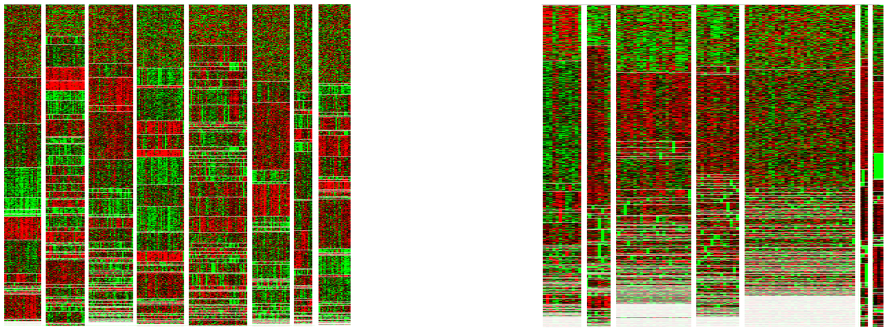


Figure 4.5: Observed protein expression arranged according to posterior estimated cluster structure under NoB-Loc. Only active proteins are displayed. Panel (a) shows the result of application of the NobLoc model on the original data and panel (b) shows the results after the application of POE.

Figure 4.6 shows one of the blocks in Figure 4.5 in more details highlighting the similarities between the cell lines and proteins in that block. Sam-

ples are reasonably homogeneous in terms of the expressions of the particular subgroup of proteins shown in the figure. Notice also that some proteins present typically high expression while others typically present low expression considering the particular group of samples.

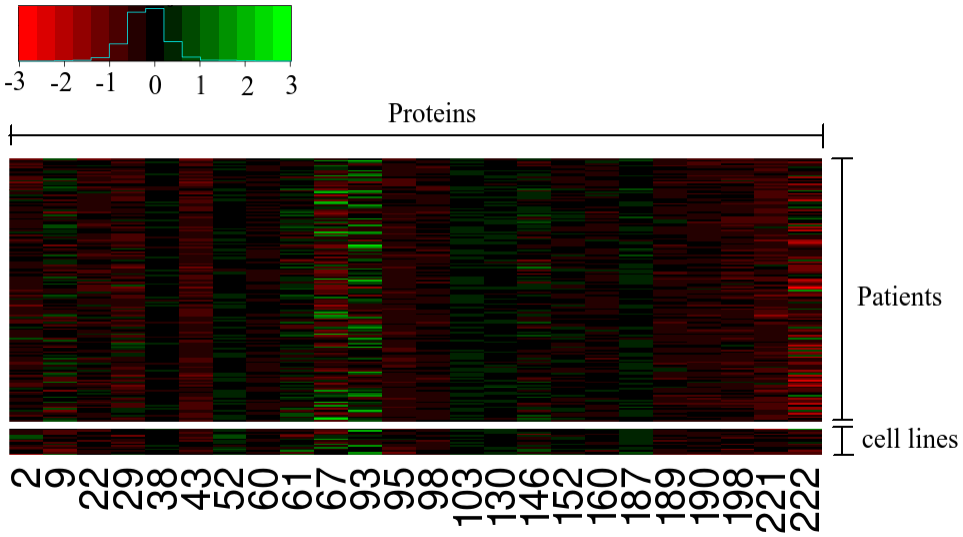


Figure 4.6: Protein expressions within one of the samples/proteins blocks of Figure 4.5 (b). The cell lines and patients exhibit very similar profiles when considering the subset of proteins that were clustered together by the model.

4.6 Discussion and Future Directions

Some innovations were introduced in this chapter: (i) The model naturally accounts for co-expression of genes and/or proteins in a manner that allows distinct clustering of samples depending on each estimated subgroup of co-expressed functional units or pathways; (ii) We develop efficient models for matching patients and cell line profiles. Development of such inferential tools

is of critical relevance to implementing precision medicine, since it can be used for potential treatment assignment that is specific to a patient, and borrows information not only from that patient, but also from cell lines coming from external sources. (iii) Third, the approach does not need to be restricted to cell lines and patients but can be generalized to multiplatform omics profiles from diverse model systems such as patient-derived xenographs (PDX) models and organoids.

This chapter makes two important methodological contributions in Bayesian non-parametrics: (i) the seamless integration of the (modified) probability of expression (POE) model for noise reduction and the nested biclustering approach; (ii) the formal probabilistic modeling of co-clustering between proteins/genes and patients based on profile similarities via dependent priors on partition models.

There are some areas that need to be more thoroughly investigated. One of them is the need to extend the simulation studies where the underlying truth differs from the POE and NobLoc models in order to address the performance of the proposed methodology under model misspecification. In our results, we restricted the matching of cell lines and patients to a conditional zero inflated Plya urn (see section 4.3.1), therefore the investigation of the other models for matching information from cell lines to patients is also proposed as a future work.

Chapter 5

Conclusions and future directions

The common theme of the three major projects in this thesis was the use of dependent priors for mixtures and random partitions. We discussed some motivating examples where the scientific research questions naturally give rise to such dependent structures. Information on a specific form of dependence represents potentially useful expert knowledge, which should be exploited when available. It is still common practice, however, to ignore such domain knowledge and proceed with default independent priors. For the specific examples discussed in this thesis we constructed suitable dependent models, developed practicable posterior inference methods and demonstrated the proposed approaches in simulation studies and in the actual applications.

Many open questions remain. For the application to cell lineage data, the approach proposed in Chapter 3 can be characterized as an empirical fit to the data with a model that respects the dependencies that arise from the nature of the data. In future research we plan to consider alternative generative models that mimick the actual biologic process of how cells diversify. A generative model is used, for example, in Shiffman et al. (2018), however still without using restrictions and informative priors that would arise from

the nature of the data.

Also for the problem of matching patients and patient clusters with representative cell lines that we considered in Chapter 4, many open questions remain. In the currently proposed model we match each patient cluster with one representative cell line, implicitly allowing each cell line to be paired with only one patient cluster. This restriction could be removed, giving rise to a slightly different random structure. Another aspect of the problem is that investigators have a preference for reporting very distinct clusters of proteins (genes) and similarly for the nested clusters of patients. That is, the desired summary of the random partition should perhaps take into account preferences for parsimony and interpretability. And such preferences could alternatively already be included in the prior probability model. One approach is the use of repulsive prior probability models that favor very distinct clusters, for example the determinantal point process (DPP) (Xu et al., 2016). Similar issues arise with the applications in Chapters 2 and 3.

Appendices

Appendix A

Probability distributions

Here we describe the parameterization used for some of the probability distributions referred throughout the text. Namely: Gamma, Inverse Gamma, Exponential, Laplace, univariate and multivariate Student T (with location-scale parameters), univariate and multivariate Gaussian.

A.1 Normal

A continuous random variable X follows a normal distribution $N(\mu, \sigma^2)$ if its density is

$$f_X(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{1}{2}\left(\frac{x-\mu}{2\sigma}\right)^2\right\}, \quad x \in \mathbb{R}.$$

It follows that $E(X) = \mu$ and $Var(X) = \sigma^2$.

A.2 Multivariate Normal

A continuous random vector $\boldsymbol{X} \in \mathbb{R}^d$ follows a $d-$variate normal distribution $N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ if its density is

$$f_{\boldsymbol{X}}(\boldsymbol{x}) = (2\pi)^{-\frac{d}{2}} \det|\Sigma|^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}(\boldsymbol{X}-\boldsymbol{\mu})^{\top}\boldsymbol{\Sigma}^{-1}(\boldsymbol{X}-\boldsymbol{\mu})\right\}, \quad \boldsymbol{x} \in \mathbb{R}^d,$$

for $\boldsymbol{\Sigma}$ positive definite. It follows that $E(\boldsymbol{X}) = \boldsymbol{\mu}$ and $Var(\boldsymbol{X}) = \boldsymbol{\Sigma}^2$.

A.3 Gamma

A continuous random variable X follows a Gamma distribution Gama(a, b) if its density is

$$f_X(x) = \frac{b^a}{\Gamma(a)} x^{a-1} e^{-bx}, \quad x > 0.$$

It folows that $E(X) = \frac{a}{b}$ and $Var(X) = \frac{a}{b^2}$.

A.4 Inverse Gamma

A continuous random variable X follows an Inverse Gamma distribution with parameters a and b, i. e., $X \sim GamaInv(a, b)$ if the random variable $Y = 1/X$ follows $Gamma(a, b)$. Then, X has density

$$f_X(x) = \frac{b^a}{\Gamma(a)} x^{-a-1} e^{-\frac{b}{x}}, \quad x > 0.$$

It follows that $E(X) = \frac{b}{a-1}$ and $Var(X) = \frac{b^2}{(a-1)^2(a-2)}$.

A.5 Univariate Student-T

A continuous random variable X follows a Stident T distribution with ν degrees of freedom, location μ and scale σ, if its density is

$$f_X(x) = \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma\sqrt{\nu\pi}\Gamma\left(\frac{\nu}{2}\right)}\left[\left(\frac{x-\mu}{\sigma}\right)^2 + \nu\right]^{-\frac{\nu+1}{2}}, \quad x \in \mathbb{R}.$$

In this case, we denote $X \sim T(\nu, \mu, \sigma)$. Under this parameterization, we have $E(X) = \mu$ if $\nu > 1$ and $Var(X) = \sigma^2 \times \frac{\nu}{\nu-2}$ for $\nu > 2$. For $\nu = 1$, the average of the Student-T is not defined and if $\nu \leq 2$, the same holds for the variance.

A.6 Multivariate Student-T

A continuous random vector $\boldsymbol{X} \in \mathbb{R}^d$ follows a multivariate Student-T distribution with ν degrees of freedom and parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ if its density is

$$f_{\boldsymbol{X}}(\boldsymbol{x}) \propto \left[d + (\boldsymbol{x}-\boldsymbol{\mu})'\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right]^{-\frac{n+d}{2}} \quad \boldsymbol{x} \in \mathbb{R}^d.$$

In this case, we denote $\boldsymbol{X} \sim T_n(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. Here $\boldsymbol{\mu}$ is the location parameter and the positive definite matrix $\boldsymbol{\Sigma}$ is the scale matrix. In this parameterization, $E(\boldsymbol{X}) = \boldsymbol{\mu}$ if $\nu > 1$ and $Var(\boldsymbol{X}) = \boldsymbol{\Sigma} \times \frac{\nu}{\nu-2}$ for $\nu > 2$. Similar to the unidimensional case, $for \nu = 1$ the mean of the distribution is not defined and if $\nu \leq 2$, the same happens for Σ.

An important result states that if $\boldsymbol{X} \sim T_n(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, then the marginals are Student are also Student-T: $X_i \sim T_n(\mu_i, \sigma_i^2)$ where μ_i is the i-th entry of the mean vector and $\sigma_i^2 = \boldsymbol{\Sigma}_{i,i}$.

A.7 Laplace

A continuous random variable $X \in \mathbb{R}^+$ follows a Laplace distribution with parameter λ, location μ and scale $\sigma > 0$ if its density is

$$f_X(x) = \frac{\lambda}{2\sigma} \exp\left\{-\frac{\lambda|y-\mu|}{\sigma}\right\}, \quad x > 0.$$

In this case, we denote $X \sim \mathit{Laplace}(\lambda, \mu, \sigma)$. It follows that $X \sim \mathit{Laplace}(\lambda, \mu, \sigma) \Rightarrow E(X) = \mu$ and $Var(X) = 2\sigma^2$.

A.8 Negative Binomial

A disctrete random variable X follows a Negative Binomial distribution with parameters $n \in \mathbb{R}^+$ e $p \in (0, 1)$ if X its probability mass function

$$p_X(x) = \frac{\Gamma(x+n)}{\Gamma(x+1)\Gamma(n)}(1-p)^n p^x, \quad x = 0, 1, 2, \ldots$$

In this case, we denote $X \sim \text{NegBin}(n, p)$.

Under such parameterization, $E(X) = \frac{np}{1-p}$ and $Var(X) = \frac{np}{(1-p)^2}$.

A.9 Log Normal

A continuous random variable X follows a LogNormal(μ, σ) distribution if the random variable $Y := \log X$ follows a normal distribution $N(\mu, \sigma^2)$. In this case, the density function of X is

$$f_X(x) = \frac{1}{x\sqrt{2\pi\sigma^2}} \exp\left\{-\left(\frac{\log x - \mu}{\sigma}\right)^2\right\}, \quad x > 0.$$

It follows that $E(X) = e^{\mu + \frac{\sigma^2}{2}}$ and $Var(X) = (e^{\sigma^2} - 1)e^{2\mu+\sigma^2}$.

Appendix B

Appendix for Chapter 2

B.1 Full Conditionals

We briefly describe the full conditional posterior distributions, numbered (1) through (9) below, that define the transition probabilities in the Gibbs sampler MCMC implementation. We define $\boldsymbol{\Psi} := (\boldsymbol{\theta}, \boldsymbol{y})$ as the random vector that includes the full parameter vector as well as the data, and we use the notation $\boldsymbol{\Psi}_{-a}$ to represent $\boldsymbol{\Psi}$ excluding component a.

1. **Updating** v_{0u}, $u = 1, 2, 3$:

$$(v_{0u} \mid \boldsymbol{\Psi}_{-v_{0u}}) \sim \text{Gamma}\left(a_v + \frac{1}{2}CDL \times \kappa_u, \right. \\ \left. b_v + \frac{1}{2}\sum_{c=1}^{C}\sum_{d=1}^{D}\sum_{\ell=1}^{L}\sum_{m=1}^{\kappa_u}(\mu^*_{cd\ell u}(m) - \mu_{0u})^2\right).$$

2. **Updating** μ_{0u}, $u = 1, 2, 3$:

$$(\mu_{0u} \mid \boldsymbol{\Psi}_{-\mu_{0u}}) \sim N\left(\frac{v_{0u}\sum_{c=1}^{C}\sum_{d=1}^{D}\sum_{\ell=1}^{L}\sum_{m=1}^{\kappa_u}\mu^*_{cd\ell u}(m) + \mu_{00}v_{00}}{v_{00} + CDL \times \kappa_u v_{0u}}, \right. \\ \left. \frac{1}{v_{00} + CDL \times \kappa_u v_{0u}}\right).$$

3. **Updating** $\mu^*_{cd\ell u}$**:** From equation (1), the vector $\boldsymbol{\mu}_{cd\ell i}$ can be written as

$$\boldsymbol{\mu}_{cd\ell i} = \mu^*_{cd\ell 1}\left(\delta^1_{cdi}\right)\boldsymbol{u}_1 + \mu^*_{cd\ell 2}\left(\delta^2_{cdi}\right)\boldsymbol{u}_2 + \mu^*_{cd\ell 3}\left(\delta^1_{cdi}\right)\boldsymbol{u}_3,$$

where $\boldsymbol{u}_1 = (1, \ldots, 1, 0, \ldots, 0)^\top$, $\boldsymbol{u}_2 = (0, \ldots, 0, 1, \ldots, 1, 0, \ldots, 0)^\top$ and $\boldsymbol{u}_3 = (0, \ldots, 0,$
$1, \ldots, 1)^\top$ with 1's in positions $1, \ldots, \tau^1_{cd\ell}$ (for $\boldsymbol{u}_1$), in positions $\tau^1_{cd\ell} + 1 \ldots \tau^2_{cd\ell}$ (for $\boldsymbol{u}_2$) and in position $\tau^2_{cd\ell} + 1 \ldots T$ (for $\boldsymbol{u}_3$), respectively. We find that $(\mu^*_{cd\ell 1}(m) \mid \boldsymbol{\Psi}_{-\mu^*_{cd\ell 1}(m)}) \sim N(a_1, b_1)$, with $b_1 = (v_{01} + J(\#\mathcal{P}^m_{cd1})\boldsymbol{u}_1^\top \boldsymbol{\Sigma}_c^{-1}\boldsymbol{u}_1)^{-1}$ and

$$a_1 = b_1 \left[\left(\sum_{i \in \mathcal{P}^m_{cd1}} \sum_j \boldsymbol{y}_{cd\ell ij} - J\boldsymbol{u}_2 \sum_{i \in \mathcal{P}^m_{cd1}} \mu^*_{cd\ell 2}(\delta^2_{cdi}) - J(\#\mathcal{P}^m_{cd1})\boldsymbol{u}_3 \mu^*_{cd\ell 3}(m) \right)^\top \boldsymbol{\Sigma}_c^{-1}\boldsymbol{u}_1 + \mu_{01} v_{01} \right],$$

where $\mathcal{P}^m_{cd1} := \{1 \le i \le I : \delta^1_{cdi} = m\}$.
Similarly, $(\mu^*_{cd\ell 2}(m) \mid \boldsymbol{\Psi}_{-\mu^*_{cd\ell 2}(m)}) \sim N(a_2, b_2)$, with
$b_2 = (v_{02} + J(\#\mathcal{P}^m_{cd2})\boldsymbol{u}_2^\top \boldsymbol{\Sigma}_c^{-1}\boldsymbol{u}_2)^{-1}$ and

$$a_2 = b_2 \left[\left(\sum_{i \in \mathcal{P}^m_{cd2}} \sum_j \boldsymbol{y}_{cd\ell ij} - J\boldsymbol{u}_1 \sum_{i \in \mathcal{P}^m_{cd2}} \mu^*_{cd\ell 1}(\delta^1_{cdi}) - J\boldsymbol{u}_3 \sum_{i \in \mathcal{P}^m_{cd\ell 2}} \mu^*_{cd\ell 3}(\delta^1_{cdi}) \right)^\top \boldsymbol{\Sigma}_c^{-1}\boldsymbol{u}_2 + \mu_{02} v_{02} \right],$$

where $\mathcal{P}^m_{cd2} := \{1 \le i \le I : \delta^2_{cdi} = m\}$.
And $(\mu^*_{cd\ell 3}(m) \mid \boldsymbol{\Psi}_{-\mu^*_{cd\ell 3}(m)}) \sim N(a_3, b_3)$, with
$b_3 = (v_{03} + J(\#\mathcal{P}^m_{cd1})\boldsymbol{u}_3^\top \boldsymbol{\Sigma}_c^{-1}\boldsymbol{u}_3)^{-1}$ and

$$a_3 = b_3 \left[\left(\sum_{i \in \mathcal{P}^m_{cd1}} \sum_j \boldsymbol{y}_{cd\ell ij} - J\boldsymbol{u}_2 \sum_{i \in \mathcal{P}^m_{cd1}} \mu^*_{cd\ell 2}(\delta^2_{cdi}) - J(\#\mathcal{P}^m_{cd1})\boldsymbol{u}_1 \mu^*_{cd\ell 1}(m) \right)^\top \boldsymbol{\Sigma}_c^{-1}\boldsymbol{u}_3 + \mu_{03} v_{03} \right].$$

4. **Updating $\boldsymbol{\Sigma}_c$:** Under the normal-inverse Wishart conjugate model we get

$$(\boldsymbol{\Sigma}_c \mid \boldsymbol{\Psi}_{\_\boldsymbol{\Sigma}_c}) \sim IW \left(IDLJ + \nu_\Sigma, \sum_{i=1}^{I}\sum_{d=1}^{D}\sum_{\ell=1}^{L}\sum_{j=1}^{J} (\boldsymbol{y}_{cdi\ell j} - \boldsymbol{\mu}_{cd\ell i})(\boldsymbol{y}_{cdi\ell j} - \boldsymbol{\mu}_{cd\ell i})^\top + V_{\Sigma_c} \right)$$

5. **Updating γ:**

$$(\gamma \mid \boldsymbol{\Psi}_{-\gamma}) \sim \text{Beta} \left(a_\gamma + \sum_{c=1}^{C}\sum_{d=1}^{D}\sum_{i=1}^{I} \mathbb{1}(\delta^2_{cdi} = \delta^1_{cdi}), b_\gamma + \sum_{c=1}^{C}\sum_{d=1}^{D}\sum_{i=1}^{I} \mathbb{1}(\delta^2_{cdi} \neq \delta^1_{cdi}) \right)$$

6. **Updating $\tau^1_{cd\ell}$ and $\tau^2_{cd\ell}$:** We update $\tau^1_{cd\ell}$ and $\tau^2_{cd\ell}$ in different blocks of the Gibbs sampler. This way we evaluate fewer scenarios than in the case of sampling both together in a single step, due to the restriction $\tau^1_{cd\ell} < \tau^2_{cd\ell}$.

$$p(\tau^1_{cd\ell} \mid \boldsymbol{\Psi}_{-\tau^1_{cd\ell}}) \propto \prod_{i=1}^{I}\prod_{j=1}^{J} N(\boldsymbol{y}_{cd\ell ij}; \boldsymbol{\mu}_{cd\ell i}, \boldsymbol{\Sigma}_c), \quad \tau^1_{cd\ell} < \tau^2_{cd\ell}.$$

$$p(\tau^2_{cd\ell} \mid \boldsymbol{\Psi}_{-\tau^2_{cd\ell}}) \propto \prod_{i=1}^{I}\prod_{j=1}^{J} N(\boldsymbol{y}_{cd\ell ij}; \boldsymbol{\mu}_{cd\ell i}, \boldsymbol{\Sigma}_c), \quad \tau^1_{cd\ell} < \tau^2_{cd\ell}. \tag{B.1}$$

If evaluating the probabilities in (B.1) is too computationaly intensive, one can alternatively implement a Metropolis-Hastings transition probability, proposing unit increments or decrements, subject to the constraint $\tau^1_{cd\ell} < \tau^2_{cd\ell}$. This would require at most four evaluations of the right hand side product in (B.1).

7. **Updating cluster membership indicators δ^1_{cdi}:** If $\delta^2_{cdi} \leq \kappa_1$, then δ^1_{cdi} is equal to the value of δ^2_{cdi} with probability 1. Otherwise, by multinomial-Dirichlet conjugacy results, the full conditional distribution of δ^1_{cdi} is $P(\delta^1_{cdi} = m \mid \boldsymbol{\Psi}_{-\delta^1_{cdi}}) \propto \mathcal{N}^1_{cdi}(m)\pi^1_m$, $m = 1, \ldots, \kappa_1$ where $\mathcal{N}^1_{cdi}(m) = \prod_\ell \prod_j N(\boldsymbol{y}_{cd\ell ij} \mid \boldsymbol{\mu}_{cd\ell i}, \boldsymbol{\Sigma}_c)$ with $\boldsymbol{\mu}_{cd\ell i}$ evaluated under $\delta^1_{cdi} = m$.

8. **Updating cluster membership indicators δ^2_{cdi}:** The full conditional p.m.f. for δ^2_{cdi} is given by

$$P(\delta^2_{cdi} = m \mid \boldsymbol{\Psi}_{-\delta^2_{cdi}}) \propto \begin{cases} \gamma \times \mathcal{N}^2_{cdi}(\delta^1_{cdi}) & \text{if } m = \delta^1_{cdi}. \\ (1-\gamma) \times \pi^2_{m-\kappa_1} \mathcal{N}^2_{cdi}(m) & \text{if } \kappa_1 + 1 \leq m \leq \kappa_2. \end{cases}$$

where $\mathcal{N}^2_{cdi}(m) = \prod_\ell \prod_j N(\boldsymbol{y}_{cd\ell ij} \mid \boldsymbol{\mu}_{cd\ell i}, \boldsymbol{\Sigma}_c)$ with $\boldsymbol{\mu}_{cd\ell i}$ being calculated assuming $\delta^2_{cdi} = m$.

9. **Updating $\boldsymbol{\pi}_1$ and $\boldsymbol{\pi}_2$:** under the conjugate multinomial-Dirichlet model we find the following posterior distribution. Let $n^1_m = \sum_{c=1}^C \sum_{d=1}^D \sum_{i=1}^I \mathbb{1}(\delta^1_{cdi} = m)$. for $m = 1, \ldots, \kappa_1$. Similarly, let $n^2_m = \sum_{c=1}^C \sum_{d=1}^D \sum_{i=1}^I \mathbb{1}(\delta^2_{cdi} = m)$ for $m = \kappa_1 + 1, \ldots, \kappa_1 + \kappa_2$. Then $(\boldsymbol{\pi}_1 \mid \boldsymbol{\Psi}_{-\boldsymbol{\pi}_1}) \sim \text{Dir}(\eta_{11}+n^1_1, \ldots, \eta_{1\kappa_1}+n^1_{\kappa_1})$ and $(\boldsymbol{\pi}_2 \mid \boldsymbol{\Psi}_{-\boldsymbol{\pi}_2}) \sim \text{Dir}(\eta_{21}+n^2_{\kappa_1+1}, \ldots, \eta^2_{\kappa_2-\kappa_1}+ n^2_{\kappa_1+\kappa_2})$.

B.2 Number of Model Parameters for AIC and BIC

Here we describe how the number of parameters was determined when evaluating the BIC criterion in sections 4 and 5 and AIC in section 4. The description focuses on BIC, but the same arguments are valid for evaluation of AIC.

The number of parameters for a given model is a function of κ_1 and κ_2 that can be decomposed as $N(\kappa_1, \kappa_2) = f(\kappa_1, \kappa_2) + const$, where *const* depends on the number of data points, but not on κ_1 or κ_2. The only parameters in the likelihood that vary in number as κ_1 and κ_2 change are $\{\boldsymbol{\mu}^*_{\ell,u} : c \in [C], d \in [D], \ell \in [L], u \in [3]\}$, which contains $f(\kappa_1, \kappa_2) = CDL(2\kappa_1 + \kappa_2)$ parameters. Therefore, $BIC = 2 \log p(\boldsymbol{y} \mid \theta) - N(\kappa_1, \kappa_2) \log n$, where n is the number of observations, hence the comparison of any pair of models is invariant with respect to the term *const* and we can, for simplicity, consider $N(\kappa_1, \kappa_2) = CDL(2\kappa_1 + \kappa_2)$.

Appendix C

Appendix for Chapter 3

C.1 Proper Prior on $(\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k, b_1, \ldots, b_k, k)$

Here we show that $p(\mu_1, \ldots, \mu_k, b, k)$ defined in (3.2) is a proper prior. In fact, since

$$\exp\left\{-\alpha \sum_{j=1}^{k} d(\mu_j, \mu_{b_j})\right\} < 1, \quad \forall\ 1 \leq j \leq k,$$

it follows that

$$Z_k \leq \int \cdots \int_{\mathbb{R}^{p\times k}} \prod_{j=1}^{k} [p(\mu_j)]\, k^k \; d\mu_1 \ldots d\mu_k = k^k < \infty.$$

Without loss of generality, we can truncate $P(k)$ to have suport $1 \leq k \leq M$ for some finite upper limit M and therefore $p(k)$ will be also proper. The truncation is justified in practical appications since one expects finite number of nodes in the tree.

C.2 Full Conditional Distributions for the s-MST Model

We now describe posterior inference under the model (3.1) - (3.10). A simple MCMC can be implemented in such a scenario, leading to the Gibbs

sampling transition probabilities:

1. Updating c_i:

$$\begin{aligned} p(c_i = k | y_i, \mu_k, \Sigma, w_k) &\propto L(y_i | c_i = k, \mu_k, \Sigma)\ p(c_i = k | w_k) \\ &\propto w_k\ \mathcal{N}(y_i; \mu_k, \Sigma), \quad \forall k = 0, \ldots, k. \end{aligned}$$

2. Updating w:

$$\begin{aligned} p(w | c_1, \ldots, c_n) &\propto p(c_1, \ldots, c_n | w)\ p(w) \\ &\sim \text{Dirichlet}(n_0 + \delta, \ldots, n_k + \delta). \end{aligned}$$

3. Updating Σ:

$$\begin{aligned} p(\Sigma | Y, \text{rest}) &\propto L(Y | \mu_k, c_1, \ldots, c_n, \Sigma)\ p(\Sigma) \\ &\propto \prod_{i=1}^{n} \left\{ |\Sigma|^{-\frac{1}{2}} e^{-\frac{1}{2}(x_i - \mu_{c_i})^T \Sigma^{-1} (x_i - \mu_{c_i})} \right\} |\Sigma|^{-\frac{\nu+p+1}{2}} e^{-\frac{1}{2}\text{tr}(R\Sigma^{-1})} \\ &\sim \text{Inv-Wishart} \left(\nu + n, \Psi + \sum_i (y_i - \mu_{c_i})(y_i - \mu_{c_i})^T \right). \end{aligned}$$

4. Updating $\mu_k, \forall k = 1, \ldots, k$:

$$\begin{aligned} p(\mu_k | Y, \text{rest}) &\propto L(Y | c_1, \ldots, c_n, \mu_k, \Sigma)\ p(\mu_j | \mu^{(-j)}, b, \alpha, k) \\ &\sim \mathcal{N} \left((\Sigma_p^{-1} + n_j \Sigma^{-1})^{-1} \left[\Sigma_p^{-1} \mu_p + \Sigma^{-1} \sum_{i: c_i = j} x_i \right], \right. \\ &\qquad \left. (\Sigma_p^{-1} + n_j \Sigma^{-1})^{-1} \right), \end{aligned}$$

where f_j, μ_p, Σ_p were defined in (3.5).

5. Updating the branching structure: by resampling it from the prior conditionals $p(b_j = i|\mu_0, \ldots, \mu_k, b^{(-j)}, k)$ described in (3.3).

6. Updating the dimension K using a RJ-MCMC move:

   [label=()]Generate a proposal $\tilde{k} \sim q(\tilde{k}|k)$ and a matching set of parameters $\tilde{\theta}_{\tilde{k}} \sim p_1(\tilde{\theta}_{\tilde{k}}|y')$ as described in Section 3.3.1 Accept $(\tilde{k}, \tilde{\theta}_{\tilde{k}})$ with probability α defined in (3.8).

C.3 Full Conditional Distributions for the h-MST Model

First, we list the full conditional distributions for implementation of Gibbs sampler on the model described in Section 3.2.2 conditionaly on the dimension k.

(a) Updating c_j:

$$\begin{aligned} p(c_i = j|y_i, \mu_j, \Sigma_j, w_j, k) &\propto L(y_i|c_i = j, \mu_j, \Sigma)\ p(c_i = j|w_j) \\ &\propto w_j\ \mathcal{N}(y_i; \mu_j, \Sigma_j), \quad \forall j = 0, \ldots, k. \end{aligned}$$

2. Updating w:

$$\begin{aligned} p(w|c_1, \ldots, c_n, k) &\propto p(c_1, \ldots, c_n|w, k)\ p(w \mid k) \\ &\sim \text{Dirichlet}(n_0 + \delta, \ldots, n_k + \delta). \end{aligned}$$

3. Updating Σ^{-1}

$$p(\Sigma^{-1} \mid \boldsymbol{y}, rest) \propto \prod_{i=1}^{n} p(\boldsymbol{y}_i \mid \boldsymbol{\mu}_{c_i}, \Sigma^{-1}) p(\Sigma^{-1})$$
$$\sim Wishart\left(n+\nu, \left[\Psi^{-1} + \sum_{i}^{n} (\boldsymbol{y}_i - \boldsymbol{\mu}_{c_i})(\boldsymbol{y}_i - \boldsymbol{\mu}_{c_i})^{\top}\right]^{-1}\right).$$

4. Updating $\boldsymbol{\mu}_j$:

   The conditional posterior density $p(\boldsymbol{\mu}_j \mid \boldsymbol{y}, \text{rest})$ is not straightforward to either write in analytic form or to sample from.

   Denote by $S_j := \{i : \ c_i = j\}$ the set of observations that belong to cluster j. Combining the likelihood with the h-MST prior, we have

$$p(\boldsymbol{\mu}_j \mid \boldsymbol{\mu}^{(-j)}, \boldsymbol{y}, rest) \propto \left[\prod_{i \in S_j} N(\boldsymbol{y}_i; \boldsymbol{\mu}_j, \Sigma) N(\boldsymbol{\mu}_j; \boldsymbol{m}, \sigma_0^2 I)\right] \times$$
$$\times \exp\left\{-\alpha \mathcal{W}(MST(\boldsymbol{\mu}_1, \cdots, \boldsymbol{\mu}_k))\right\},$$

   in which the sum $\mathcal{W}(MST(\boldsymbol{\mu}_1, \cdots, \boldsymbol{\mu}_k))$ involves different terms depending on the position of $\boldsymbol{\mu}_j$ in $\mathbb{R}^D$. This leads to the full conditional being a finite mixture of truncated normals, with non-overlapping truncation regions $A_l \subset \mathbb{R}^D,\ l = 1, \ldots, n$ such that the neighbors of any node $\boldsymbol{\mu}_j \in A_l$ are the same under the $MST(\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k)$ when we fix the remaining nodes $\boldsymbol{\mu}^{(-j)}$. The challenge lies in defining the regions A_l that partition $\mathbb{R}^D$.

However, we can build a tractable and efficient Metropolis Hastings proposal $q(\tilde{\boldsymbol{\mu}}_j \mid \boldsymbol{\mu}_j; \boldsymbol{\mu}^{(-j)}$,

$\boldsymbol{y}, rest)$ to approximate $p(\boldsymbol{\mu}_j \mid \boldsymbol{y}, \text{rest})$. We will omit the dependence on variables other than $\boldsymbol{\mu}_j$ from the notation for clarity of exposition, therefore writing $q(\tilde{\boldsymbol{\mu}}_j \mid \boldsymbol{\mu}_j)$ instead of $q(\tilde{\boldsymbol{\mu}}_j \mid \boldsymbol{\mu}_j; \boldsymbol{\mu}^{(-j)}, \boldsymbol{y}, rest)$. We define $q(\tilde{\boldsymbol{\mu}}_j \mid \boldsymbol{\mu}_j)$ as follows. Take from the edges $E_{\boldsymbol{\mu}_1,\ldots,\boldsymbol{\mu}_k}$ of the $MST(\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k)$ the subset $V_j = \{i : \{j, i\} \in E_{\boldsymbol{\mu}_1,\ldots,\boldsymbol{\mu}_k}\}$ of all neighbors of node j. We propose a new component specific mean $\tilde{\boldsymbol{\mu}}_j$ from

$$q(\tilde{\boldsymbol{\mu}}_j \mid \boldsymbol{\mu}_j) \propto \prod_{i \in S_j} N(\boldsymbol{y}_i; \tilde{\boldsymbol{\mu}}_j, \Sigma) N(\tilde{\boldsymbol{\mu}}_j; \boldsymbol{m}, \sigma_0^2 I) \exp\left\{-\alpha \sum_{i \in V_j} d^2(\boldsymbol{\mu}_i, \tilde{\boldsymbol{\mu}}_j)\right\},$$

which simplifies to $q(\tilde{\boldsymbol{\mu}}_j \mid \boldsymbol{\mu}_j) \sim N(\tilde{\boldsymbol{\mu}}_j;\ \boldsymbol{a}_j, \boldsymbol{B}_j)$, where $\boldsymbol{B}_j = (|S_j|\Sigma^{-1} + \sigma_0^{-2} I + 2\alpha |V_j| I)^{-1}$ and $\boldsymbol{a}_j = \boldsymbol{B}_j \left(\Sigma^{-1} \sum_{i \in S_j} \boldsymbol{y}_i + \sigma_0^{-2} \boldsymbol{m} + 2\alpha \sum_{l \in V_j} \boldsymbol{\mu}_l\right)$. By denoting the proposed neighborhood of $\tilde{\boldsymbol{\mu}}_j$ as $\tilde{V}_j = \{i : \{i, j\} \in E_{\boldsymbol{\mu}^*_1,\ldots,\boldsymbol{\mu}^*_k}\}$ where $\boldsymbol{\mu}^*_l = \boldsymbol{\mu}_l$ if $l \neq j$ and $\boldsymbol{\mu}^*_j = \tilde{\boldsymbol{\mu}}_j$, the resulting Metropolis Hastings acceptance probability equals 1 if $\tilde{V}_j = V_j$ and

$$\begin{aligned}
\alpha(\tilde{\boldsymbol{\mu}}_j \mid \boldsymbol{\mu}_j) &= \min\left\{1, \frac{q(\boldsymbol{\mu}_j \mid \tilde{\boldsymbol{\mu}}_j) p(\tilde{\boldsymbol{\mu}}_j \mid \boldsymbol{y})}{q(\tilde{\boldsymbol{\mu}}_j \mid \boldsymbol{\mu}_j) p(\boldsymbol{\mu}_j \mid \boldsymbol{y}))}\right\} \\
&= \min\left\{1,\ \frac{\left[\prod_{i \in S_j} N(\boldsymbol{y}_i; \tilde{\boldsymbol{\mu}}_j, \Sigma_j)\right] N(\tilde{\boldsymbol{\mu}}_j; \boldsymbol{m}, \sigma_0^2 I)}{\left[\prod_{i \in S_j} N(\boldsymbol{y}_i; \boldsymbol{\mu}_j, \Sigma_j)\right] N(\boldsymbol{\mu}_j; \boldsymbol{m}, \sigma_0^2 I)} \times \right. \\
&\qquad \left. \times \frac{\exp\left\{-\alpha \mathcal{W}(MST(\tilde{\boldsymbol{\mu}}_j, \boldsymbol{\mu}^{(-j)}))\right\}}{\exp\left\{-\alpha \mathcal{W}(MST(\boldsymbol{\mu}_1, \cdots, \boldsymbol{\mu}_k))\right\}} \times \frac{N(\boldsymbol{\mu}_j;\ \tilde{\boldsymbol{a}}_j, \tilde{\boldsymbol{B}}_j)}{N(\tilde{\boldsymbol{\mu}}_j;\ \boldsymbol{a}_j, \boldsymbol{B}_j)}\right\}
\end{aligned}$$

$$= \min \left\{ 1, \quad \frac{\left[\prod_{i \in S_j} N(\boldsymbol{y}_i; \tilde{\boldsymbol{\mu}}_j, \Sigma_j)\right] N(\tilde{\boldsymbol{\mu}}_j; \boldsymbol{m}, \sigma_0^2 I)}{\left[\prod_{i \in S_j} N(\boldsymbol{y}_i; \boldsymbol{\mu}_j, \Sigma_j)\right] N(\boldsymbol{\mu}_j; \boldsymbol{m}, \sigma_0^2 I)} \frac{N(\boldsymbol{\mu}_j;\ \tilde{\boldsymbol{a}}_j, \tilde{\boldsymbol{B}}_j)}{N(\tilde{\boldsymbol{\mu}}_j;\ \boldsymbol{a}_j, \boldsymbol{B}_j)} \times \right.$$
$$\left. \times \frac{\exp\left\{-\alpha \sum_{i \in \tilde{V}_j} d^2(\tilde{\boldsymbol{\mu}}_j, \boldsymbol{\mu}_i)\right\}}{\exp\left\{-\alpha \sum_{i \in V_j} d^2(\boldsymbol{\mu}_j, \boldsymbol{\mu}_i)\right\}} \right\}$$

otherwise.

Appendix D

Appendix for Chapter 4

D.1 Full Conditionals for the POE Model

Here we briefly describe how to sample from each one of the full conditionals. In this section, we use $\mathbb{1}(\cdot)$ to denote either an indicator function or th support of a truncated probability distribution. We also define the sets $\mathcal{P}_g^+ := \{1 \leq t \leq T : \ e_{sg} = 1\}$ and $\mathcal{P}_t^+ := \{1 \leq g \leq G : \ e_{sg} = 1\}$. The sets $\mathcal{P}_g^0,\ \mathcal{P}_g^-,\ \mathcal{P}_t^0$ and $\mathcal{P}_t^-$ are defined analogously.

Updating e_{sg}

$$p(e_{sg} = x \mid y, else) \propto \begin{cases} \frac{\pi_g^+}{k_g^+} \times \mathbb{1}(\alpha_s + \mu_g < y_{sg} < \alpha_s + \mu_g + k_g^+), & \text{if } x = 1, \\ \pi_g^0 \times N(y_{sg};\ \alpha_s + \mu_g,\ \sigma_g^2), & \text{if } x = 0, \\ \frac{\pi_g^-}{k_g^-} \times \mathbb{1}(\alpha_s + \mu_g - k_g^- < y_{sg} < \alpha_s + \mu_g), & \text{if } x = -1. \end{cases}$$

Updating k_g^+ and k_g^-

$$(k_g^+ \mid y, else) \sim InvGamma(\#\mathcal{P}_g^+ + \alpha_{k^+},\ \beta_{k^+}) \\ \mathbb{1}\left(k_g^+ > \max\left(\max_{t\in\mathcal{P}_g^+}\{y_{sg} - \alpha_s - \mu_g\},\ k_0\sigma_g\right)\right).$$

$$(k_g^- \mid y, else) \sim InvGamma(\#\mathcal{P}_g^- + \alpha_{k^-},\ \beta_{k^-}) \\ \mathbb{1}\left(k_g^- > \max\left(\max_{t\in\mathcal{P}_g^-}\{\alpha_s + \mu_g - y_{sg}\},\ k_0\sigma_g\right)\right).$$

Updating $\boldsymbol{\pi}_g$

$$(\boldsymbol{\pi}_g \mid y, else) \sim Dirichlet\left((\ \#\mathcal{P}_g^- + \alpha_\pi^-,\ \#\mathcal{P}_g^0 + \alpha_\pi^0,\ \#\mathcal{P}_g^+ + \alpha_\pi^+)\right).$$

Updating σ_g^2

$$(\sigma_g^2 \mid y, else) \sim InvGamma\left(\frac{\#\mathcal{P}_g^0}{2} + \gamma,\ \frac{1}{2}\sum_{t\in\mathcal{P}_g^0}(y_{sg} - \alpha_s - \mu_g)^2 + \lambda\right) \\ \mathbb{1}\left(\sigma_g^2 < \frac{\min\left(k_g^+, k_g^-\right)^2}{k_0^2}\right).$$

Updating μ_g

$$(\mu_g \mid y, else) \sim N(a_g, b_g)\mathbb{1}\left(\max\left(M_g^+, M_g^-\right) < \mu_g < \min\left(m_g^+, m_g^-\right)\right),$$

where

$$
\begin{aligned}
M_g^+ &= \max\{y_{sg} - \alpha_s;\ t \in \mathcal{P}_g^+\} - k_g^+; \\
M_g^- &= \max\{y_{sg} - \alpha_s;\ t \in \mathcal{P}_g^-\}; \\
m_g^+ &= \min\{y_{sg} - \alpha_s;\ t \in \mathcal{P}_g^+\}; \\
m_g^+ &= \min\{y_{sg} - \alpha_s;\ t \in \mathcal{P}_g^-\} + k_g^-; \\
b_g &= (\#\mathcal{P}_g^0 \sigma_g^{-2} + \tau_\mu^{-1})^{-1}; \\
a_g &= b_g \times \left[\sigma_g^{-2} \sum_{t \in \mathcal{P}_g^0} (y_{sg} - \alpha_s) + \tau_\mu^{-1} \theta_\mu \right].
\end{aligned}
$$

Updating α_s

$$
(\alpha_s \mid y, else) \sim N(a_t, b_t)
$$

$$
\mathbb{1}\left(\max\left(M_t^+, M_t^-\right) < \alpha_s < \min\left(m_t^+, m_t^-\right)\right) \mathbb{1}\left(\sum_{t=1}^{T} \alpha_s = 0\right).
$$

where

$$
\begin{aligned}
M_t^+ &= \max\{y_{sg} - \mu_g - k_g^+;\ g \in \mathcal{P}_t^+\}; \\
M_t^- &= \max\{y_{sg} - \mu_g;\ g \in \mathcal{P}_t^-\}; \\
m_t^+ &= \min\{y_{sg} - \mu_g;\ g \in \mathcal{P}_t^+\}; \\
m_t^+ &= \min\{y_{sg} - \mu_g + k_g^-;\ g \in \mathcal{P}_t^-\}; \\
b_t &= \left(\sum_{g \in \mathcal{P}_t^0} \sigma_g^{-2} + \tau_\alpha^{-1} \right)^{-1}; \\
a_t &= b_t \times \left[\sum_{t \in \mathcal{P}_t^0} \left(\frac{y_{sg} - \alpha_s}{\sigma_g^2} \right) + \tau_\alpha^{-1} \mu_\alpha \right].
\end{aligned}
$$

Updating θ_μ

$$(\theta_\mu \mid y, else) \sim N\left(\left(m_\mu s_\mu^{-2} + \tau_\mu^{-1}\sum_{g=1}^{G}\mu_g\right)(s_\mu^{-2} + G\tau_\mu^{-1})^{-1},\ (s_\mu^{-2} + G\tau_\mu^{-1})^{-1}\right)$$

Updating τ_μ

$$(\tau_\mu \mid y, else) \sim InvGamma\left(\frac{G}{2} + a_{\tau_\mu},\ \frac{1}{2}\sum_{g=1}^{G}(\mu_g - \theta_\mu)^2 + b_{\tau_\mu}\right)$$

Updating β_{k^+}

$$(\beta_{k^+} \mid y, else) \sim Gamma\left(G\alpha_{k^+} + a_{\beta_{k^+}},\ b_{\beta_{k^+}} + \sum_{g=1}^{G}\frac{1}{k_g^+}\right)$$

Updating β_{k^-}

$$(\beta_{k^-} \mid y, else) \sim Gamma\left(G\alpha_{k^-} + a_{\beta_{k^-}},\ b_{\beta_{k^-}} + \sum_{g=1}^{G}\frac{1}{k_g^-}\right)$$

Updating α_{k^+}

$p(\alpha_{k^+} \mid y, else)$ is not analytically available since

$$p(\alpha_{k^+} \mid y, else) \propto \Gamma(\alpha_{k^+})^{-G}\left(\frac{\beta_{k^+}^G}{\prod_{g=1}^{G}k_g^+}\right)^{\alpha_{k^+}} e^{-\alpha_{k^+}\lambda_{\alpha_{k^+}}}.$$

One way of (approximately) sampling from this distribution is through the Metropolis-Hastings scheme.

We specify a proposal $q(\alpha_{k^+}^{new} \mid \alpha_{k^+}^{old})$ corrected by the acceptance probability $\alpha(\alpha_{k^+}^{new} \mid \alpha_{k^+}^{old}) := \min\left\{1,\ r(\alpha_{k^+}^{new} \mid \alpha_{k^+}^{old})\right\}$, where

$$r(\alpha_{k^+}^{new} \mid \alpha_{k^+}^{old}) := \frac{q(\alpha_{k^+}^{old} \mid \alpha_{k^+}^{new}) p(\alpha_{k^+}^{new} \mid y, else)}{q(\alpha_{k^+}^{new} \mid \alpha_{k^+}^{old}) p(\alpha_{k^+}^{old} \mid y, else)}.$$

Our proposal is a random-walk on $\log \alpha_{k^+}$, i.e., $\log \alpha_{k^+}^{new} \sim N(\log \alpha_{k^+}^{old}, V^+)$ for some fixed $V^+ > 0$. which implies a LogNormal proposal density on the original scale with $q(\alpha_{k^+}^{new} \mid \alpha_{k^+}^{old}) = N(\alpha_{k^+}^{new}; \alpha_{k^+}^{old}, V^+) \times 1/\alpha_{k^+}^{new}$.

It is straightforward to verify that

$$\log r(\alpha_{k^+}^{new} \mid \alpha_{k^+}^{old}) = (\log \alpha_{k^+}^{old} - \log \alpha_{k^+}^{new}) - G\left[\log \Gamma(\alpha_{k^+}^{new}) - \log \Gamma(\alpha_{k^+}^{old})\right] + \\ + (\alpha_{k^+}^{new} - \alpha_{k^+}^{old}) \left[G \log \beta_{k^+} - \sum_{g=1}^{G} \log k_g^+ - \lambda_{\alpha_{k^+}}\right].$$

Updating α_{k^-}

Updating α_{k^-} is entirely analogous to updating α_{k^+}.

D.1.1 Sampling from truncated distributions within MCMC

In this section we describe the Gibbs sampler augmentation scheme to asymptotically sample from the truncated inverse gamma and truncated normal distributions that appear in appendix D.1. Although sampling algorithms for truncated distributions can be easy derived, sometimes even by the inverse c.d.f. method, the resulting algorithm can often be numerically unstable (take

the truncated Gaussian distribution for example). In such cases, it could be advantageous to use an approximate sampler if it is more robust to computational errors. In this regard, we follow the directions on Damien and Walker (2001).

The univariate normal sampling scheme can be seen as a particular instance of the algorithm for multivariate normals or as an extension of the sampling scheme for univariate standard normals that are both described in Damien and Walker (2001). The algorithm to sample from truncated inverse gamma is very similar to the one that samples from the truncated Gamma. We describe both sampling schemes here solely for the purpose of completeness.

D.1.1.1 Truncated normal

Suppose a truncated Gaussian distribution for the random variable X: $X \sim N(\mu, \sigma^2)\mathbb{1}(a < X < b)$, i.e., $f_X(x) \propto \exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\}\mathbb{1}(a < x < b)$, where we could have $a = -\infty$ or $b = +\infty$ to represent unilateral truncation. We define the auxiliary variable Y through the joint density

$$f_{X,Y}(x, y) \propto \mathbb{1}(0 < y < e^{-\frac{(x-\mu)^2}{2\sigma^2}})\mathbb{1}(a < x < b)$$

so that the implied marginal for X matches the original $N(\mu, \sigma^2)\mathbb{1}(a < X < b)$. The full conditional distributions are

$$(Y \mid X = x) \sim Unif\left(0,\ \exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\}\right), \tag{D.1}$$

$$(X \mid Y = y) \sim Unif\left(\max(a,\ \mu - \sqrt{-2\sigma^2 \log y}),\ \ \min(b,\ \mu + \sqrt{-2\sigma^2 \log y})\right). \tag{D.2}$$

Within the MCMC scheme described in section D.1, we include sampling from the auxiliary variables Y_{μ_g} and Y_{α_t} corresponding to the full conditional distributions of μ_g, and α_t respectively. The auxiliary variables are sampled from (D.1) while the original variables are sampled from (D.2), with the appropriate values of μ, σ^2, a and b.

D.1.1.2 Truncated inverse gamma

Suppose $X \sim InvGamma(\alpha, \beta)\mathbb{1}(a < x < b)$, i.e., $f_X(x) \propto x^{-\alpha-1}e^{-\frac{x}{\beta}}$ $\mathbb{1}(a < x < b)$. We define the joint density of X and Y:

$$f_{X,Y}(x, y) \propto x^{-\alpha-1}\mathbb{1}(0 < y < e^{-\frac{x}{\beta}})\mathbb{1}(a < x < b)$$

so that the implied marginal for X matches the original $InvGamma(\alpha, \beta)\mathbb{1}(a < X < b)$. The full conditional distributions are

$$(Y \mid X = x) \sim Unif(0, e^{-\frac{x}{\beta}}), \tag{D.3}$$

$$f_{X|Y}(x \mid y) \propto x^{-\alpha-1}\mathbb{1}(M(y) < x < b), \tag{D.4}$$

where $M(y) := \max\left(a,\ -\frac{\beta}{\log y}\right)$. The inverse c.d.f. method provides an efficient way to sample from (D.4): sample $U \sim Unif(0, 1)$ then evaluate the transformed variable

$$\frac{M(y)}{[U(\{M(y)/b\}^{\alpha}-1)+1]^{\frac{1}{\alpha}}},$$

which will be distributed as (D.4).

Within the MCMC scheme described in section D.1, we include sampling from the auxiliary variables $Y_{k_g^+}$, $Y_{k_g^-}$ and $Y_{\sigma_g^2}$ corresponding to the full conditional distributions of k_g^+, k_g^- and σ_g^2 respectively. The auxiliary variables are sampled from (D.4) while the original variables are sampled from (D.3), with the appropriate values of α and β.

D.2 Full Conditionals for Matching Cell Line and Patients Model

This appendix describes steps of the Gibbs sampler algorithm used to carry out posterior inference.

Define $S_j^{x,k} := \{i : \delta_i^{x,k} = j\}$ with x being either c or p. In the remainder of this appendix section, we will denote by s_j the single element in the set $S_j^{c,k}$ (it could even be $s_j = \emptyset$), omitting the superscripts for simplicity.

The full posterior (up to a normalizing constant depending solely on the data $\boldsymbol{d}$) can be factorized as

$$p(\boldsymbol{\Psi} \mid \boldsymbol{d}) \propto \left[\prod_{g=1}^{G} p(\sigma_g^{-2}) p(\sigma_{1g}^{-2}) p(\sigma_{2g}^{-2})\right] p(\boldsymbol{w} \mid \pi_0, \alpha_0) \left[\prod_{k=1}^{K_{\boldsymbol{w}}} p(\boldsymbol{\delta}^{p,k}) p(\boldsymbol{\delta}^{c,k} \mid \boldsymbol{\delta}^{p,k})\right] \times$$
$$\times \prod_{k=1}^{K_{\boldsymbol{w}}} \prod_{g:w_g=k} \prod_{j=1}^{J_k} p(\theta_{jg}^* \mid \mu_{0g}, \sigma_{0g}^2) \times \left[\prod_{g=1}^{G} p(\mu_{0g}) p(\mu_{1g}) p(\mu_{2g})\right] \times$$
$$\times \prod_{k=1}^{K_{\boldsymbol{w}}} \prod_{g:w_g=k} \prod_{j=1}^{J_k} \left[p(d_{ig}^c \mid \theta_{jg}^*, \sigma_g^2)^{\mathbb{1}(S_j^{c,k}=\{i\}\neq\emptyset)} \prod_{i:\delta_i^{p,k}=j} p(d_{ig}^p \mid \theta_{jg}^*, \sigma_g^2)\right] \times$$
$$\times \prod_{k=1}^{K_{\boldsymbol{w}}} \prod_{g:w_g=k} \left[\prod_{i:\delta_i^{c,k}=0} p(d_{ig}^c \mid \mu_{1g}, \sigma_g^2, \sigma_{1g}^2) \prod_{i:\delta_i^{p,k}=0} p(d_{ig}^p \mid \mu_{1g}, \sigma_g^2, \sigma_{1g}^2)\right] \times$$
$$\times \prod_{g:w_g=0} \left[\prod_{i=1}^{N^p} p(d_{ig}^p \mid \mu_{2g}, \sigma_g^2, \sigma_{2g}^2) \prod_{i=1}^{N^c} p(d_{ig}^c \mid \mu_{2g}, \sigma_g^2, \sigma_{2g}^2)\right]. \tag{D.5}$$

Updating θ_{jg}^*:

$$p(\theta_{jg}^* \mid \boldsymbol{d}, \boldsymbol{\Psi}_{-\theta_{jg}^*}) \propto N(\theta_{jg}^*;\ \mu_{0g}, \sigma_{0g}^2) \prod_{i \in S_j^{c,w_g}} N(d_{ig}^c;\ \theta_{jg}^*, \sigma_g^2) \prod_{i \in S_j^{c,w_g}} N(d_{ig}^p;\ \theta_{jg}^*, \sigma_g^2)$$

$$(\theta_{jg}^* \mid \boldsymbol{d}, \boldsymbol{\Psi}_{-\theta_{jg}^*}) \sim N\left(\frac{(\sum_{i \in S_j^{c,w_g}} d_{ig}^c + \sum_{i \in S_j^{p,w_g}} d_{ig}^p)\sigma_g^{-2} + \mu_{0g}\sigma_{0g}^{-2}}{(|S_j^{c,w_g}| + |S_j^{p,w_g}|)\sigma_g^{-2} + \sigma_{0g}^{-2}}, \right.$$
$$\left. \frac{1}{(|S_j^{c,w_g}| + |S_j^{p,w_g}|)\sigma_g^{-2} + \sigma_{0g}^{-2}}\right).$$

Updating $\boldsymbol{\delta}^{p,k}$:

We follow Bush and MacEachern (1996) and sample the cluster membership indicators $\delta^{p,k}$ within a Gibbs block marginalizing $\boldsymbol{\theta}^*$ out, i.e., by

sampling $p(\delta_i^{p,k} = j \mid \boldsymbol{d}, \boldsymbol{\Psi}_{-(\delta_i^{p,k}, \boldsymbol{\theta}^*)})$ for $i = 1, \ldots, N^p$. These updates together with the previous one where we sampled $\theta_{jg}^* \sim p(\theta_{jg}^* \mid \boldsymbol{\Psi}_{-\theta_{jg}^*})$ for all j and g, asymptotically provides a blocked joint sample from $p(\boldsymbol{\theta}^*, \boldsymbol{\delta}^{p,k} \mid \boldsymbol{d}, \boldsymbol{\Psi}_{-(\boldsymbol{\delta}^{p,k}, \boldsymbol{\theta}^*)})$.

Denote by $A_{p,k}^-$ the number of active patient samples within protein sample k excluding patient i and define $S_j^{p,k-} := S_j^{p,k} \setminus \{i\}$. Then $\boldsymbol{\delta}^{p,k} \sim ZEPU(\alpha_{p,k}, \pi_{pk})$ implies

$$P(\delta_i^{p,k} = j \mid \boldsymbol{\delta}_{-i}^{p,k}) = \begin{cases} \pi_{pk}, & j = 0 \\ (1 - \pi_{pk}) \frac{|S_j^{p,k-}|}{\alpha_{pk} + A_{p,k}^-}, & j = 1, \ldots, J_k^- \\ (1 - \pi_{pk}) \frac{\alpha_{p,k}}{\alpha_{pk} + A_{p,k}^-}, & j = J_k^- + 1. \end{cases} \tag{D.6}$$

Using equation (D.6), we obtain

$$\begin{aligned} P(\delta_i^{p,k} = j \mid \boldsymbol{\delta}^{c,k}, \boldsymbol{\delta}_{-i}^{p,k}) &\propto p(\boldsymbol{\delta}^{c,k} \mid \boldsymbol{\delta}^{p,k}) P(\delta_i^{p,k} = j \mid \boldsymbol{\delta}_{-i}^{p,k}) \\ &\propto P(\delta_i^{p,k} = j \mid \boldsymbol{\delta}_{-i}^{p,k}) \sum_{n=0}^{\min\{J_k, N^c\}} \binom{J_k}{n} \binom{N^c}{n} n! \end{aligned} \tag{D.7}$$

analytically. Notice that J_k varies with $\delta_i^{p,k}$ so the summation term cannot by omitted from (D.7).

After marginalizing θ_{jg}^* out from $p(d_{ig}^p \mid \boldsymbol{d}_{-ig}^p, d_{s_j g}^c, \delta_i^{p,k} = j, \boldsymbol{\Psi}_{-\delta_i^{p,k}})$, we obtain

$$
\begin{aligned}
&p(d_{ig}^p \mid \boldsymbol{d}_{-ig}^p, d_{s_j g}^c, \delta_i^{p,k} = j, \boldsymbol{\Psi}_{-(\delta_i^{p,k}, \boldsymbol{\theta}^*)}) \\
&= \sqrt{\frac{(|S_j^{p,k-}|\sigma_g^{-2} + \sigma_{0g}^{-2})(|S_j^{p,k-}|\sigma_g^{-2} + \sigma_{0g}^{-2} + \sigma_g^{-2})}{(2\pi\sigma_g^2)}} \times \\
&\times \exp\left\{ -\frac{1}{2} \left[{d_{ig}^p}^2 \sigma_g^{-2} + (|S_j^{p,k-}|\sigma_g^{-2} + \sigma_{0g}^{-2}) \times \right.\right. \\
&\qquad \left.\left. \times \left(\sigma_g^{-2} \sum_{\ell \in S_j^{p,k-}} d_{\ell g}^p + d_{s_j g}^c \mathbb{1}(S_j^{c,k} \neq \emptyset) \sigma_g^{-2} + \mu_{0g} \sigma_{0g}^{-2} \right)^2 \right] \right\} \times \\
&\times \exp\left\{ \frac{1}{2} (\sigma_g^{-2} + \sigma_{0g}^{-2} + |S_j^{p,k-}|\sigma_g^{-2})^{-1} \times \right. \\
&\qquad \left. \times \left[d_{ig}^p \sigma_g^{-2} + \left(\sum_{\ell \in S_j^{p,k-}} d_{\ell g}^p + d_{s_j g}^c \mathbb{1}(S_j^{c,k} \neq \emptyset) \right) \sigma_g^{-2} + \mu_{0g} \sigma_{0,g}^{-2} \right] \right\}.
\end{aligned}
$$

Using equation (D.7), we get

$$
\begin{aligned}
&P(\delta_i^{p,k} = j \mid \boldsymbol{\Psi}_{-(\delta_i^{p,k}, \boldsymbol{\theta}^*)}) \propto \\
&\propto \begin{cases}
\left[\prod_{g:w_g=k} N(d_{ig}^p \mid \mu_{1g}, \sigma_g^2 + \sigma_{1g}^2) \right] \pi_{pk} \sum_{n=0}^{J_k^- + 1} \frac{1}{(N^c - n)!} b_j, & j = 0 \\
\prod_{g:w_g=k} p(d_{ig}^p \mid \boldsymbol{d}_{-ig}^p, d_{s_j g}^c, \delta_i^{p,k} = j, \boldsymbol{\Psi}_{-(\delta_i^{p,k}, \boldsymbol{\theta}^*)}) \times & \\
\qquad \times (1 - \pi_{pk}) \frac{|S_j^{p,k} - \{i\}|}{\alpha_{pk} + A_{p,k}^-} \sum_{n=0}^{J_k^- + 1} \frac{1}{(N^c - n)!} b_j, & j = 1, \ldots, J_k^- \\
\prod_{g:w_g=k} N(d_{ig}^p \mid \mu_{0g}, \sigma_g^2 + \sigma_{0g}^2) \times & \\
\qquad \times (1 - \pi_{pk}) \frac{\alpha_{p,k}}{\alpha_{pk} + A_{p,k}^-} \sum_{n=0}^{J_k^- + 2} \frac{1}{(N^c - n)!} b_j, & j = J_k^- + 1,
\end{cases}
\end{aligned}
$$

where J_k^- is the number of active clusters of patients within protein group k after removing patient i and $b_j := \mathbb{1}(|S_j^{c,k}| = 1)\mathbb{1}(|S_j^{p,k}| > 0) + \mathbb{1}(|S_j^{c,k}| = 0)$ is a binary variable that enforces non-empty active clusters of cell lines to contain at least one patient sample as well.

Updating $\boldsymbol{\delta}_i^{c,k}$:

$$
\begin{aligned}
&p(\delta_i^{c,k} = j \mid \boldsymbol{d}, \boldsymbol{\Psi}_{-\delta_i^{c,k}}) \propto \\
&\quad \propto \begin{cases} \prod_{g:w_g=k} N(d_{ig}^c;\ \theta_{jg}^*, \sigma_{1g}^2), & j > 0,\ S_j^{p,k} \neq \emptyset,\ S_j^{c,k} = \emptyset \\ \prod_{g:w_g=k} N(d_{ig}^c;\ \mu_1, \sigma_g^2 + \sigma_{1g}^2), & j = 0 \\ 0, & \text{otherwise.} \end{cases}
\end{aligned}
$$

We also define a Metropolis-Hastings algorithm to sample from $\boldsymbol{\delta}^{c,k}$ in a way that hopefully produces Markov Chains with better mixing properties. Here we omit the upper indexes c, k from $\boldsymbol{\delta}^{c,k}$ for clarity of exposition.

Recall that the Metropolis-Hastings algorithm produces a new sample $\boldsymbol{\delta}^{t+1}$ from $\boldsymbol{\delta}^t$ according to an auxiliary transition probability $q(\boldsymbol{\delta}^{t+1} \mid \boldsymbol{\delta}^t)$ that is irreducible and aperiodic. Then $\boldsymbol{\delta}^{t+1}$ is accepted with probability $\alpha(\boldsymbol{\delta}^{t+1} \mid \boldsymbol{\delta}^t) := \max\{1, r(\boldsymbol{\delta}^{t+1} \mid \boldsymbol{\delta}^t)\}$ where $r(\boldsymbol{\delta}^{t+1} \mid \boldsymbol{\delta}^t) := \frac{q(\boldsymbol{\delta}^t|\boldsymbol{\delta}^{t+1})p(\boldsymbol{\delta}^{t+1})}{q(\boldsymbol{\delta}^{t+1}|\boldsymbol{\delta}^t)p(\boldsymbol{\delta}^t)}$.

We define two types of transitions and at each iteration we uniformly chose one of them at random.

Type I: We take an active cell line and switch it with one of the inactive cell lines. Under such proposal, $q(\boldsymbol{\delta}^{t+1} \mid \boldsymbol{\delta}^t) = \frac{1}{|S_0^{c,k}|(N^c - |S_0^{c,k}|)}$. Under this proposal, the acceptance ratio reduces to

$$r(\boldsymbol{\delta}^{c,k}(t+1) \mid \boldsymbol{\delta}^{c,k}(t)) = \prod_{g:w_g=k} \frac{N(d^c_{i_0(t+1)}; \mu_1, \sigma_g^2 + \sigma_{1g}^2) N(d^c_{ig}; \theta^*_{jg}, \sigma_g^2)\big|_{j=\delta^{c,k}_{i_1(t+1)}}}{N(d^c_{i_0(t)}; \mu_1, \sigma_g^2 + \sigma_{1g}^2) N(d^c_{ig}; \theta^*_{\ell g}, \sigma_g^2)\big|_{\ell=\delta^{c,k}_{i_1(t)}}},$$

where $i_1(x)$ and $i_0(x)$ respectively denote the active and inactive cell lines selected by the proposal at time x (before switching, when $x = t$; and after switching, when $x = t+1$).

Type II: Randomly pick an active cluster j. If $|S_j^{c,k}(t)| = 0$ (no active cell line in cluster j), assign an inactive cell line to cluster j uniformly at random. On the other hand, if $|S_j^{c,k}(t)| = 1$ we reassign the only active cell line $i \in S_j^{c,k}(t)$ from cluster j to the group of inactive cell lines by making $\delta_i^{c,k}(t+1) = 0$. Under such proposal, we have $q(\boldsymbol{\delta}^{t+1} \mid \boldsymbol{\delta}^t) = \frac{1}{J_k |S_0^{c,k}(t)|} \mathbb{1}(|S_j^{c,k}| = 0) + \frac{1}{J_k} \mathbb{1}(|S_j^{c,k}| = 1)$. Under this proposal,

$$r(\boldsymbol{\delta}^{c,k}(t+1) \mid \boldsymbol{\delta}^{c,k}(t)) = \begin{cases} \frac{N(d^c_{ig};\ \theta^*_{jg}, \sigma_g^2) |S_0^{c,k}(t)|}{N(d^c_{ig};\ \mu_1, \sigma_{0g}^2 \sigma_g^2)}\Big|_{i \in S_j^{c,k}(t+1)}, & S_j^{c,k}(t) = \emptyset \\ \frac{N(d^c_{ig};\ \mu_1, \sigma_{0g}^2 \sigma_g^2)}{N(d^c_{ig};\ \theta^*_{jg}, \sigma_g^2) |S_0^{c,k}(t+1)|}\Big|_{i \in S_j^{c,k}(t)}, & S_j^{c,k}(t) = \{i\} \neq \emptyset. \end{cases}$$

Bibliography

Akaike, H. (1973). Maximum likelihood identification of Gaussian autoregressive moving average models. *Biometrika*, 60(2):255–265.

Alston, C. L., Strickland, C. M., Mengersen, K. L., and Gardner, G. E. (2012). Bayesian mixed effects models. *Case Studies in Bayesian Statistical Modelling and Analysis*, pages 141–158.

Azadi, S., Brush, R. S., Anderson, R. E., and Rajala, R. V. (2016). Class I phosphoinositide 3-kinase exerts a differential role on cell survival and cell trafficking in retina. In *Retinal Degenerative Diseases: Mechanisms and Experimental Therapy*, pages 363–369, Cham. Springer International Publishing.

Boruvka, O. (1926). Contribution to the solution of a problem of economical construction of electrical networks. *Elektronickỳ Obzor*, 15:153–154.

Bush, C. A. and MacEachern, S. N. (1996). A semiparametric Bayesian model for randomised block designs. *Biometrika*, 83(2):275–285.

Caron, F., Neiswanger, W., Wood, F., Doucet, A., and Davy, M. (2017). Generalized Pólya urn for time-varying Pitman-Yor processes. *Journal of Machine Learning Research*, 18(27):1–32.

Caunt, C. J., Sale, M. J., Smith, P. D., and Cook, S. J. (2015). MEK1 and MEK2 inhibitors and cancer therapy: the long and winding road. *Nature Reviews Cancer*, 15(10):577–592.

Charboneau, L., Scott, H., Chen, T., Winters, M., Petricoin, E. F., Liotta, L. A., et al. (2002). Utility of reverse phase protein arrays: applications to signalling pathways and human body arrays. *Briefings in Functional Genomics and Proteomics*, 1(3):305–315.

Chung, Y. and Dunson, D. B. (2011). The local Dirichlet process. *Annals of the Institute of Statistical Mathematics*, 63(1):59–80.

Cristinelli, S. and Ciuffi, A. (2018). The use of single-cell RNA-Seq to understand virus–host interactions. *Current opinion in virology*, 29:39–50.

Dahl, D. B. (2006). Model-based clustering for expression data via a Dirichlet process mixture model. *Bayesian Inference for Gene Expression and Proteomics*, pages 201–218.

Damien, P. and Walker, S. G. (2001). Sampling truncated normal, beta, and gamma densities. *Journal of Computational and Graphical Statistics*, 10(2):206–215.

Dellaportas, P. and Papageorgiou, I. (2006). Multivariate mixtures of normals with unknown number of components. *Statistics and Computing*, 16(1):57–68.

Dharampuriya, P. R., Scapin, G., Wong, C., Wagner, K. J., Cillis, J. L., and Shah, D. I. (2017). Tracking the origin, development, and differentiation of hematopoietic stem cells. *Current opinion in cell biology*, 49:108–115.

Elliott, L. T., De Iorio, M., Favaro, S., Adhikari, K., Teh, Y. W., et al. (2018). Modeling population structure under hierarchical Dirichlet processes. *Bayesian Analysis*. Advance publication, DOI: 10.1214/17-BA1093.

Engelman, J. A. (2009). Targeting PI3K signalling in cancer: opportunities, challenges and limitations. *Nature Reviews Cancer*, 9(8):550.

Ferguson, T. S. (1973). A bayesian analysis of some nonparametric problems. *Annals of Statistics*, 1(2):209–230.

Ferguson, T. S., Phadia, E. G., and Tiwari, R. C. (1992). Bayesian Nonparametric Inference. *Lecture Notes-Monograph Series*, 17:127–150.

Fletcher, R. B., Das, D., Gadye, L., Street, K. N., Baudhuin, A., Wagner, A., Cole, M. B., Flores, Q., Choi, Y. G., Yosef, N., et al. (2017). Deconstructing olfactory stem cell trajectories at single-cell resolution. *Cell Stem Cell*, 20(6):817–830.

Frühwirth-Schnatter, S. (2006). *Finite mixture and Markov switching models*. Springer Science & Business Media.

Frühwirth-Schnatter, S., Celeux, G., and Robert, C. (2019). *Handbook of Mixtures*. CRC Press.

Gelman, A., Hwang, J., and Vehtari, A. (2014). Understanding predictive information criteria for Bayesian models. *Statistics and Computing*, 24(6):997–1016.

Goodspeed, A., Heiser, L. M., Gray, J. W., and Costello, J. C. (2016). Tumor-derived cell lines as molecular models of cancer pharmacogenomics. *Molecular Cancer Research*, 14(1):3–13.

Green, P. J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82(4):711–732.

Kingman, J. F. C. (1978). Random partitions in population genetics. *Proceedings of the Royal Society. London. Series A. Mathematical, Physical and Engineering Sciences*, 361(1704):1–20.

Korthauer, K. D., Chu, L.-F., Newton, M. A., Li, Y., Thomson, J., Stewart, R., and Kendziorski, C. (2016). A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. *Genome biology*, 17(1):222.

Kulesza, A. and Taskar, B. (2012). Determinantal point processes for machine learning. *Foundations and Trends in Machine Learning*, 5(23):123–286.

Lachos, V. H., Castro, L. M., and Dey, D. K. (2013). Bayesian inference in nonlinear mixed-effects models using normal independent distributions. *Computational Statistics & Data Analysis*, 64:237–252.

Lau, J. W. and Green, P. J. (2007). Bayesian model-based clustering procedures. *Journal of Computational and Graphical Statistics*, 16(3):526–558.

Lee, J., Müller, P., Gulukota, K., Ji, Y., et al. (2015). A Bayesian feature allocation model for tumor heterogeneity. *The Annals of Applied Statistics*, 9(2):621–639.

Lee, J., Müller, P., Zhu, Y., and Ji, Y. (2013). A nonparametric Bayesian model for local clustering with application to proteomics. *Journal of the American Statistical Association*, 108(503):775–788.

Lindstrom, M. J. and Bates, D. M. (1990). Nonlinear mixed effects models for repeated measures data. *Biometrics*, pages 673–687.

Liu, P., Cheng, H., Roberts, T. M., and Zhao, J. J. (2009). Targeting the phosphoinositide 3-kinase pathway in cancer. *Nature Reviews Drug Discovery*, 8(8):627.

Mardia, K. V., Kent, J. T., and Bibby, J. M. (1979). *Multivariate Analysis*. London, Academic Press.

Miragaia, R. J., Teichmann, S. A., and Hagai, T. (2017). Single-cell insights into transcriptomic diversity in immunity. *Current Opinion in Systems Biology*, 5:63–71.

Neal, R. (2003). Density modeling and clustering using Dirichlet diffusion trees. In Bernardo, J., Bayarri, M. J., Dawid, A. P., Berger, J. O., Heckerman, D., Smith, A. F. M., and West, M., editors, *Bayesian Statistics 7*, pages 619–629, Oxford, UK. Oxford University Press.

Nitulescu, G., Margina, D., Juzenas, P., Peng, Q., Olaru, O., Saloustros, E., et al. (2016). AKT inhibitors in cancer treatment: the long journey from drug discovery to clinical use (review). *International Journal of Oncology*, 48(3):869–885.

O'Hagan, A. (1995). Fractional Bayes factors for model comparison. *Journal of the Royal Statistical Society. Series B (Methodological)*, 57(1):99–138.

Park, T. and Casella, G. (2008). The Bayesian lasso. *Journal of the American Statistical Association*, 103(482):681–686.

Parmigiani, G., Garrett, E. S., Anbazhagan, R., and Gabrielson, E. (2002). A statistical framework for expression-based molecular classification in cancer. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 64(4):717–736.

Perraudeau, F., Risso, D., Street, K., Purdom, E., and Dudoit, S. (2017). Bioconductor workflow for single-cell rna sequencing: Normalization, dimensionality reduction, clustering, and lineage inference. *F1000Research*, 6.

Pitman, J. and Yor, M. (1987). The two-parameter Poisson-Dirichlet distribution derived from a stable subordinator. *The Annals of Probability*, 25:855–900.

Podsypanina, K., Lee, R. T., Politis, C., Hennessy, I., Crane, A., Puc, J., et al. (2001). An inhibitor of mTOR reduces neoplasia and normalizes p70/S6

kinase activity in PTEN$^{+/-}$ mice. *Proceedings of the National Academy of Sciences*, 98(18):10320–10325.

Prim, R. C. (1957). Shortest connection networks and some generalizations. *Bell Labs Technical Journal*, 36(6):1389–1401.

Richardson, S. and Green, P. J. (1997). On Bayesian analysis of mixtures with an unknown number of components (with discussion). *Journal of the Royal Statistical Society: series B (statistical methodology)*, 59(4):731–792.

Risso, D., Perraudeau, F., Gribkova, S., Dudoit, S., and Vert, J.-P. (2018). A general and flexible method for signal extraction from single-cell RNA-seq data. *Nature Communications*, 9(1):284.

Robert, C. and Casella, G. (2013). *Monte Carlo Statistical Methods*. Springer Science & Business Media.

Rodríguez, A. and Ter Horst, E. (2008). Bayesian dynamic density estimation. *Bayesian Analysis*, **3**:339–366.

Rousseau, J. and Mengersen, K. (2011). Asymptotic behaviour of the posterior distribution in overfitted mixture models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(5):689–710.

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464.

Serra, V., Scaltriti, M., Prudkin, L., Eichhorn, P. J., Ibrahim, Y. H., Chandarlapaty, S., et al. (2011). PI3K inhibition results in enhanced HER signaling and acquired ERK dependency in HER2-overexpressing breast cancer. *Oncogene*, 30(22):2547.

Sethuraman, J. (1994). A constructive definition of dirichlet priors. *Statistica sinica*, pages 639–650.

Shiffman, M., Stephenson, W. T., Schiebinger, G., Huggins, J., Campbell, T., Regev, A., and Broderick, T. (2018). Reconstructing probabilistic trees of cellular differentiation from single-cell RNA-seq data. *arXiv e-prints*, arXiv:1811.11790.

Sinha, R., Schultz, N., and Sander, C. (2015). Comparing cancer cell lines and tumor samples by genomic profiles. *bioRxiv*, page 028159.

Spiegelhalter, D. J., Best, N. G., Carlin, B. P., and Van Der Linde, A. (2002). Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 64(4):583–639.

Street, K., Risso, D., Fletcher, R. B., Das, D., Ngai, J., Yosef, N., Purdom, E., and Dudoit, S. (2018). Slingshot: cell lineage and pseudotime inference for single-cell transcriptomics. *BMC Genomics*, 19(1):477.

Stubbington, M. J., Rozenblatt-Rosen, O., Regev, A., and Teichmann, S. A. (2017). Single-cell transcriptomics to explore the immune system in health and disease. *Science*, 358(6359):58–63.

Süli, E. and Mayers, D. F. (2003). *An introduction to numerical analysis*. Cambridge university press.

Teh, Y. W., Blundell, C., and Elliott, L. (2011). Modelling genetic variations using fragmentation-coagulation processes. In *Advances in Neural Information Processing Systems 24*, pages 819–827. Curran Associates, Inc.

Teh, Y. W., Jordan, M. I., Beal, M. J., and Blei, D. M. (2006). Hierarchical Dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566–1581.

Wade, S., Ghahramani, Z., et al. (2018). Bayesian cluster analysis: point estimation and credible balls. *Bayesian Analysis*, 13(2):526–558.

Watanabe, S. (2010). Asymptotic equivalence of Bayes cross validation and widely applicable information criterion in singular learning theory. *Journal of Machine Learning Research*, 11(Dec):3571–3594.

Wilson, N. K. and Göttgens, B. (2018). Single-cell sequencing in normal and malignant hematopoiesis. *Hemasphere*, 2(2):e34.

Xu, Y., Müller, P., and Telesca, D. (2016). Bayesian inference for latent biologic structure with determinantal point processes (dpp). *Biometrics*, 72(3):955–964.

Zanini, C. T. P., Müller, P., Ji, Y., and Quintana, F. A. (2019). A Bayesian random partition model for sequential refinement and coagulation. *Biometrics*, pages (in press), https://doi.org/10.1111/biom.13047.

Zhang, Z., Chan, K. L., Wu, Y., and Chen, C. (2004). Learning a multivariate Gaussian mixture model with the reversible jump MCMC algorithm. *Statistics and Computing*, 14(4):343–355.