Bayesian Variable Selection in Clustering High-Dimensional Data via a Mixture of Finite Mixtures

Abstract

When clustering high-dimensional data, it is often important to identify variables that discriminate the clusters. Meanwhile, a common issue in clustering problems is to determine the number of clusters. In this study, we propose a new method that simultaneously performs clustering and variable selection, while inferring the number of clusters from the data. We formulate the clustering problem using a finite mixture model with a symmetric Dirichlet weights prior, while also placing a prior on the number of components. That is, we utilize a mixture of finite mixtures. We handle the variable selection problem by introducing a latent binary vector, which represents the inclusion/exclusion of variables into the set of discriminating variables. We update the binary vector for variable selection using a Metropolis algorithm and perform inference on the cluster structure using a split–merge Markov chain Monte Carlo technique. We demonstrate the advantage of the proposed method using simulated datasets and two real DNA microarray datasets.

Keywords: Bayesian inference; Clustering; DNA microarray data; Finite mixture model; High dimensional data; Variable selection.

1 Introduction

High-dimensional data have become common in a wide range of fields such as biology, medicine, and engineering. Often, the primary goal in the analysis of such data is to cluster the observations into homogeneous groups. For example, in the analysis of DNA microarray data on thousands of genes, a common goal is to discover disease subtypes so that treatments can be tailored to specific disease subtypes of patients. When clustering high-dimensional data, it is also often of interest to determine variables that distinguish the different clusters. Considering again the DNA microarray data example, another major focus of the clustering analysis is to identify genes that best discriminate the disease subtypes, which can be useful for understanding the mechanisms of disease progression and improving diagnoses and therapeutic interventions.

There have been previous studies to address the two problems of clustering and variable selection for high-dimensional data simultaneously. Friedman and Meulman (2004) proposed a clustering procedure to cluster observations based on separate subsets of variables for different clusters. The optimal subsets of variables for each individual cluster were detected using heuristic search strategies. This approach works in conjunction with conventional distance-based hierarchical clustering algorithms, and thus suffers from uncertainty in choosing the number of clusters. Hoff et al. (2006) also identified separate subsets of variables that characterize different clusters while uncovering the cluster structure using a mixture of Gaussian distributions, where different clusters are identified by mean shifts and discriminating variables are identified by computing Bayes factors. Tadesse et al. (2005) formulated a clustering problem in terms of a finite mixture of distributions with an unknown number of components, and then introduced a binary exclusion/inclusion latent vector to identify the discriminating variables. They utilized the reversible jump Markov chain Monte Carlo (MCMC) technique to define a sampler that moves between mixture models with different numbers of components, while selecting discriminating variables using stochastic search techniques. Unlike the

methods in Friedman and Meulman (2004) and Hoff et al. (2006), this approach assumed the same subset of discriminating variables across all clusters. Although reversible jump MCMC allows the number of clusters to be inferred, designing good reversible jump moves is difficult, particularly in high-dimensional parameter spaces. To address this issue, Kim et al. (2006) proposed an alternative approach using an infinite mixture of distributions with a Dirichlet process prior on the mixture weights, while adopting a binary latent vector for variable selection as in Tadesse et al. (2005). Compared to the other methods, Kim et al. (2006)'s approach has the computational advantage in the situation where the number of observations is relatively small compared to the number of variables since it use split–merge sampling technique which updates every sample allocation in each iteration. Moreover, using Dirichlet process mixture (DPM) models allows efficient inference of the cluster structure. However, it is also known that DPMs tend to create tiny extra clusters and overestimate the number of components (Miller and Harrison, 2013). Yau and Holmes (2011) and Malsiner-Walli et al. (2016) also formulated clustering problem in terms of Bayesian mixture model, but they used sparsity priors on the component parameters in order to select variables. Both researches used Laplace prior and normal gamma prior as shrinkage prior on the mixture component means respectively.

In this paper, we focus on the analysis of high-dimensional datasets where the sample size is substantially smaller than the number of variables. By using split–merge MCMC algorithm used by Kim et al. (2006) to simultaneously process clustering and variable selection, we take computational advantage of processing high-dimensional data whose the number of variables exceeds the number of observations. Furthermore, we propose an alternative approach to Kim et al. (2006) by formulating a clustering problem in terms of a finite mixture of distributions with a prior on the number of components so that the performance of clustering and variable selection can be improved. When the prior on the mixture weights is symmetric Dirichlet, such a mixture model is called a mixture of finite mixtures (MFM) (Richardson

and Green, 1997). Compared with DPMs, MFMs tend to associate small probabilities to partitions with tiny clusters, and lead to a consistent estimation of the number of clusters (Geng et al., 2018). Moreover, Miller and Harrison (2018) showed that MFMs exhibit many of the essential properties of DPMs, such as the Chinese restaurant process, which allows an efficient implementation of MFMs. Similar to Tadesse et al. (2005) and Kim et al. (2006), we address the variable selection problem by introducing a latent binary vector that represents the inclusion/exclusion of variables. We update the latent vector for variable selection using a Metropolis algorithm, and perform inference on the cluster structure using a split–merge MCMC technique. We demonstrate the advantage of the proposed method using a simulated dataset and two famous real DNA microarray datasets: colon cancer dataset of Alon et al. (1999) and leukemia dataset of Golub et al. (1999).

We additionally make mention of previous studies which addressed the two problems of clustering and variable selection, although they did not deal with high-dimensional data: Raftery and Dean (2006) formulated clustering problem in terms of Gaussian mixture model and compared two nested variable subsets to determine whether a variable should be included in or excluded from the current model using approximate Bayes factor. Maugis et al. (2009) generalized the model of Raftery and Dean (2006) by allowing the variables that are irrelevant in clustering to be explained by only relevant variables. The subset of relevant variables is decided through a linear regression and each model is evaluated by the Bayesian information criterion. Both approaches above used on a greedy search algorithm that could be computationally too time consuming.

2 Model Formulation

2.1 Clustering via MFM Models

Let $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$ be n independent p-dimensional observations arising from a finite mixture of distributions, with each distribution representing a different cluster. The problem of clustering the n observations can be formulated in terms of K underlying probability distributions as follows:

$$f(\mathbf{x}_i|\boldsymbol{\pi}, \boldsymbol{\theta}) = \sum_{g=1}^{K} \pi_g f(\mathbf{x}_i|\boldsymbol{\theta}_g), \tag{1}$$

where $f(\mathbf{x}_i|\boldsymbol{\theta}_g)$ is the density of an observation $\mathbf{x}_i$ from the gth component, $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K)$, and $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_K)$ are component weights satisfying $\pi_g \geq 0$ and $\sum_{g=1}^{K} \pi_g = 1$. Latent variables $Z_1, \ldots, Z_n$ are introduced to assign each observation to one of the mixture components, where $Z_i = g$ if $\mathbf{x}_i$ is generated from component g. It is assumed that $Z_1, \ldots, Z_n$ are independently and identically distributed with a probability mass function (pmf) $p(Z_i = g) = \pi_g$. In this study, we consider $f(\mathbf{x}_i|\boldsymbol{\theta}_g)$ to be a multivariate normal density with mean and covariance parameters, i.e., $\boldsymbol{\theta}_g = (\boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)$.

By treating the number of components K as an unknown parameter and placing a prior p_K on it while assuming a symmetric Dirichlet prior for $\boldsymbol{\pi}$, the MFM models are represented as follows (Miller and Harrison, 2018):

$$\begin{gathered}
K \sim p_K, \text{where } p_K \text{ is a pmf on } \{1, 2, \ldots\}, \\
(\pi_1, \ldots, \pi_k) \sim \text{Dirichlet}_k(\alpha, \ldots, \alpha), \text{ given } K = k, \\
Z_1, \ldots, Z_n \overset{iid}{\sim} \boldsymbol{\pi}, \text{ given } \boldsymbol{\pi}, \\
\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_k \overset{iid}{\sim} G_0, \text{ given } K = k, \\
\mathbf{x}_i \sim f_{\boldsymbol{\theta}_{Z_i}} \text{ independently for } i = 1, \ldots, n, \text{ given } \boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K, Z_1, \ldots, Z_n,
\end{gathered}$$

where G_0 is a base measure from which $\boldsymbol{\theta}_g$ is drawn independently and identically, and α is

the concentration parameter. Here, we choose $p_K(k)$ to be a shifted Poisson distribution on positive integers, $p_K(k) = \text{Poisson}(k-1|\lambda)$, for some $\lambda > 0$. In general, Uniform prior and Possion prior have been suggested for the prior of the number of components in the finite mixture model in the case when there is no information about the number of components (Grazian et al., 2020). Compare to the Uniform distribution, the Poisson distribution is strongly biased towards low number of components and prevents the model from generating empty components. In other words, the Poisson prior acts as a penalty term favoring simpler models of fewer components, which is similar to the comparison metrics of Akaike Information Criterion and Bayesian Information Criterion (Nobile and Fearnside, 2007). This property fits well with our model which places small probabilites on partition with tiny clusters.

Although MFMs perhaps represent the most natural Bayesian approach for mixture models with an unknown number of components, they have not become as popular as DPMs. This may be explained by the computational difficulty of MFMs. The most commonly employed inference method for MFMs is reversible jump MCMC. However, it is nontrivial to design good reversible jump moves, especially in high-dimensional spaces. Meanwhile, relatively simple and generic MCMC algorithms are available for DPMs by using computationally tractable representations of the Dirichlet process, such as its exchangeable partition distribution, the Chinese restaurant process, or the stick-breaking representation.

Recently, Miller and Harrison (2018) showed that MFMs exhibit equivalent representations of DPMs. Consequently, many of the inference algorithms developed for DPMs can also be directly applied to MFMs for an efficient implementation of MFMs. For example, the MFM counterpart to the Chinese restaurant process for DPMs is expressed as follows:

- Let C_n be the partition of $\{1, \ldots, n\}$ induced by $Z_1, \ldots, Z_n$, i.e., $C_n = \{c_g : n_g > 0\}$, where $c_g = \{i : Z_i = g\}$ for $g \in \{1, 2, \ldots\}$ and n_g is the cardinality of c_g. Initialize with a single cluster consisting of element 1 alone: $C_1 = \{\{1\}\}$.

- For $n = 2, 3, \ldots,$ element n is placed in

- an existing cluster $c \in C_{n-1}$ with probability $\propto |c| + \alpha$, where $|c|$ is the size of c;
- a new cluster with probability $\propto \frac{V_n(t+1)}{V_n(t)}\alpha$, where $V_n(t) = \sum_{k=1}^{\infty} \frac{k_{(t)}}{(k\alpha)^{(n)}} p_K(k)$ and $t = |C_{n-1}|$, where $\mathrm{x}^{(m)} = \mathrm{x}(\mathrm{x}+1)\dots(\mathrm{x}+m-1)$ and $\mathrm{x}_{(m)} = \mathrm{x}(\mathrm{x}-1)\dots(\mathrm{x}-m+1)$ with $\mathrm{x}^{(0)} = 1$ and $\mathrm{x}_{(0)} = 1$.

For comparison, the Chinese restaurant process of DPMs places the nth element in an existing cluster c with probability $\propto |c|$ or a new cluster with probability $\propto \alpha$ (the scaling parameter of Dirichlet process) (Blackwell et al., 1973).

Furthermore, Miller and Harrison (2018) showed that MFMs enable the consistent inference of the number of components. In contrast, DPMs tend to create tiny extra clusters, and are not consistent for the number of components. Compared to the Chinese restaurant process for DPMs, the introduction of new clusters is slowed down by a factor of $\frac{V_n(t+1)}{V_n(t)}$ under MFMs, suppressing the generation of tiny extraneous clusters (Geng et al., 2018). This behavior of MFMs in comparison with DPMs can be understood by comparing the conditional distribution of cluster sizes given the number of clusters. Let T be the number of clusters of the random partition of $\{1, \dots, n\}$, and let $S = (S_1, \dots, S_T)$ be the vector of the cluster sizes. Then,

$$p_{MFM}(S = s|T = t) \propto\!\!\!\approx s_1^{\alpha-1} \dots s_t^{\alpha-1} \tag{2}$$

and

$$p_{DPM}(S = s|T = t) \propto s_1^{-1} \dots s_t^{-1} \tag{3}$$

where $\propto\!\!\!\approx$ is the symbol of 'approximately proportional to' (Miller and Harrison, 2018). According to Eq.(2) and Eq.(3), MFMs place small probabilities on partitions with tiny clusters by setting α larger than 1, whereas DPMs favor partitions with many tiny clusters.

2.2 Variable Selection in Clustering

In the clustering of high-dimensional data, the inclusion of unnecessary variables could obscure the recovery of the true cluster structure (Tadesse et al., 2005). To identify the relevant variables, we introduce a latent binary vector $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_p)$, as in Tadesse et al. (2005) and Kim et al. (2006), where ξ_j has the value 1 if the variable j defines a mixture distribution and 0 otherwise. Let $\mathbf{X}_{(\xi)}$ and $\mathbf{X}_{(\xi^c)}$ denote the sets of discriminating variables and remaining variables that favor a single multivariate normal density, respectively. We assume that there is no correlation between the two sets of variables. Then, we have

$$\mathbf{x}_{i(\xi)}|Z_i = g, \boldsymbol{\theta}_g, \boldsymbol{\xi} \sim \mathcal{N}(\boldsymbol{\mu}_{g(\xi)}, \boldsymbol{\Sigma}_{g(\xi)}),$$

and

$$\mathbf{x}_{i(\xi^c)}|\boldsymbol{\eta}, \boldsymbol{\Omega}, \boldsymbol{\xi} \sim \mathcal{N}(\boldsymbol{\eta}_{(\xi^c)}, \boldsymbol{\Omega}_{(\xi^c)}),$$

where $\boldsymbol{\eta}_{(\xi^c)}$ and $\boldsymbol{\Omega}_{(\xi^c)}$ denote the mean and covariance parameters for the nondiscriminating variables, respectively. The likelihood function is then given by

$$\begin{aligned} L(Z, \boldsymbol{\xi}, \boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{\Omega}|\boldsymbol{X}) = (2\pi)^{\{-n(p-p_\xi)\}/2}|\boldsymbol{\Omega}_{(\xi^c)}|^{-n/2} \exp\{-\frac{1}{2}\sum_{i=1}^{n}(\mathbf{x}_{i(\xi^c)} - \boldsymbol{\eta}_{(\xi^c)})^T\boldsymbol{\Omega}_{(\xi^c)}^{-1}(\mathbf{x}_{i(\xi^c)} - \boldsymbol{\eta}_{(\xi^c)})\} \\ \times \prod_{g=1}^{k}(2\pi)^{-(n_g p_\xi)/2}|\boldsymbol{\Sigma}_{g(\xi)}|^{-n_g/2} \exp\{-\frac{1}{2}\sum_{i\in c_g}(\mathbf{x}_{i(\xi)} - \boldsymbol{\mu}_{g(\xi)})^T\boldsymbol{\Sigma}_{g(\xi)}^{-1}(\mathbf{x}_{i(\xi)} - \boldsymbol{\mu}_{g(\xi)})\}, \end{aligned} \tag{4}$$

where $p_\xi = \sum_{j=1}^{p}\xi_j$ and $c_g = \{i : Z_i = g\}$ with cardinality n_g.

2.3 Prior Specifications

We assume conjugate priors for the mean and covariance parameters. We also assume independence among the nondiscriminating variables, and set $\boldsymbol{\Omega} = \sigma^2 I_{p\times p}$ for computational

convenience. We set the prior distributions as follows:

$$\begin{aligned}
\boldsymbol{\mu}_{g(\xi)}|\boldsymbol{\Sigma}_{g(\xi)} &\sim \mathcal{N}(\boldsymbol{\mu}_{0(\xi)}, h_1\boldsymbol{\Sigma}_{g(\xi)}),\\
\boldsymbol{\eta}_{(\xi^c)}|\boldsymbol{\Omega}_{(\xi^c)} &\sim \mathcal{N}(\boldsymbol{\mu}_{0(\xi^c)}, h_0\boldsymbol{\Omega}_{(\xi^c)}),\\
\boldsymbol{\Sigma}_{g(\xi)} &\sim \mathcal{IW}(\delta, \boldsymbol{Q}_{1(\xi)}),\\
\sigma^2 &\sim \mathcal{IG}(a, b),
\end{aligned} \tag{5}$$

where $\mathcal{IW}(\delta, \boldsymbol{Q}_{1(\xi)})$ is the inverse-Wishart distribution with dimension p_ξ, the shape parameter $\delta = n - p_\xi + 1$, and n degrees of freedom; $\boldsymbol{Q}_{1(\xi)} = k_1 I_{p_\xi \times p_\xi}$ is the $p_\xi \times p_\xi$ positive-definite scale matrix; and $\mathcal{IG}(a, b)$ is an inverse gamma distribution with shape parameter a and scale parameter b. Many previous researches on bayesian mixture models, assumed hierarchical prior on the component specific parameters (Richardson and Green, 1997; Yau and Holmes, 2011). This hierarchical structure provides a flexibility when there is a weak prior information about the cluster structure. However, since deriving the posterior distribution is difficult when using a hierarchical prior, the computational advantage from the split-merge algorithm and collapsed gibbs algorithm is lost in this structure. Therefore, even though we could use hierarchical priors for the component parameters, we set data-driven priors which set $\boldsymbol{\mu}_{0(\xi)}$ and $\boldsymbol{\mu}_{0(\xi^c)}$ to the corresponding covariate interval midpoints. This ensure that the prior distributions overlap with the likelihood and that we can obtain well-behaved posterior densities (Wasserman, 2000). For computational convenience of marginalization, we use conjugate priors for the mean and covariance parameters instead of using semi-conjugate priors whose component mean does not depend on the component variance. This can be justified with an intuition that it is hard to pin down the mean of the data if it has large spread (Murphy, 2012). Even though the dependency of priors implies that more compact components are closer to the overall mean, the impact will not be significant since we are dealing with clustering problem of sparse high-dimensional data.

We assume independent Bernoulli distributions for the elements ξ_j of the variable selec-

tion indicator $\boldsymbol{\xi}$, as follows:

$$p(\boldsymbol{\xi}) = \prod_{j=1}^{p} \omega^{\xi_j}(1-\omega)^{1-\xi_j},$$

where ω can be determined according to prior knowledge on the expected proportion of discriminating variables to all variables.

3 Inference

3.1 Model Fitting

We iteratively update the variable selection index using repeated Metropolis steps (*Step* 1) and perform inference on the cluster structure using the split–merge algorithm (Jain and Neal, 2004)(*Step* 2), similar to Kim et al. (2006). In split–merge algorithm, component labels of observations are updated by splitting a cluster or by merging two clusters. Specifically, if two observation, which were picked at random, belong to the same component, split-procedure is proposed: all observations in the component are divided in to two components. If observations are in different components, merge-procedure is proposed: all observations are placed in to same component. Finally, whether the proposed component label is accepted is determined by Metropolis-Hastings algorithm.

While Gibbs sampler updates the component label one by one and can get stuck in local modes, The split–merge algorithm can escape from local modes by non-incrementally updating sample allocations. As demonstrated in Dahl (2003) and Jain and Neal (2004), split–merge algorithm is suitable for mixture models of high–dimensional data. The split–merge sampler is followed by a full Gibbs sampling scan (*Step* 3), so that we can exploit non-incremental (major) changes from the split–merge step and incremental (minor) refinements from the Gibbs sampling step.

Step 1. **Variable selection step:**

1. A new candidate $\boldsymbol{\xi}^{new}$ is generated by randomly choosing one of the following transition moves:

   (a) randomly pick one of p indices in $\boldsymbol{\xi}$ and changing its value;

   (b) choose two $\boldsymbol{\xi}$s, one from 0 and one from 1 independently and randomly, and switching their values.

2. The new candidate is accepted with probability

$$\min\left\{1, \frac{f(\boldsymbol{\xi}^{new}|\boldsymbol{X},\boldsymbol{Z})}{f(\boldsymbol{\xi}^{old}|\boldsymbol{X},\boldsymbol{Z})}\right\},$$

where

$$\begin{aligned} & f(\boldsymbol{\xi}|\boldsymbol{X},Z) \propto f(\boldsymbol{X}|\boldsymbol{\xi},Z)p(\boldsymbol{\xi}) \\ & = \pi^{-np/2}\prod_{g=1}^{k}\left\{H_{g(\xi)}|\boldsymbol{Q}_{1(\xi)}|^{(\delta+p_\xi-1)/2}|\boldsymbol{Q}_{1(\xi)}+\boldsymbol{S}_{g(\xi)}|^{-(n_g+\delta+p_\xi-1)/2}\right\} H_{0(\xi^c)}(\boldsymbol{S}_{0(\xi^c)})^{-(a+n/2)} \\ & \times \prod_{j=1}^{p}\omega^{\xi_j}(1-\omega)^{1-\xi_j}, \end{aligned} \tag{6}$$

with

$$\begin{aligned} H_{g(\xi)} &= (h_1 n_g+1)^{-p_\xi/2}\prod_{j=1}^{p_\xi}\frac{\Gamma\{(n_g+\delta+p_\xi-j)/2\}}{\Gamma\{(\delta+p_\xi-j)/2\}}, \\ H_{0(\xi^c)} &= (h_0 n+1)^{-(p-p_\xi)/2}b^{a(p-p_\xi)}\prod_{j=1}^{p-p_\xi}\frac{\Gamma(a+n/2)}{\Gamma(a)}, \\ \boldsymbol{S}_{g(\xi)} &= \sum_{i\in c_g}(\mathbf{x}_{i(\xi)}-\overline{\mathbf{x}}_{g(\xi)})(\mathbf{x}_{i(\xi)}-\overline{\mathbf{x}}_{g(\xi)})^T+\frac{n_g}{h_1 n_g+1}(\boldsymbol{\mu}_{0(\xi)}-\overline{\mathbf{x}}_{g(\xi)})(\boldsymbol{\mu}_{0(\xi)}-\overline{\mathbf{x}}_{g(\xi)})^T \\ \boldsymbol{S}_{0(\xi^c)} &= \prod_{j=1}^{p-p_\xi}\left[b+\frac{1}{2}\{\sum_{i=1}^{n}(\mathbf{x}_{ij(\xi^c)}-\overline{\mathbf{x}}_{j(\xi^c)})^2+\frac{n}{h_0 n+1}(\mu_{0j(\xi^c)}-\overline{\mathbf{x}}_{j(\xi^c)})^2\}\right], \end{aligned} \tag{7}$$

where $\overline{\mathbf{x}}_{g(\xi)}$ is the sample mean of discriminating variables for the cluster g, $\mathbf{x}_{ij(\xi^c)}$ is the jth nondiscriminating variable of the sample i, $\overline{\mathbf{x}}_{j(\xi^c)}$ is the sample mean of the jth

nondiscriminating variable, and $\mu_{0j(\xi^c)}$ is the jth element of $\boldsymbol{\mu}_{0(\xi)}$.

3. Update the candidate by repeating the above Metropolis step κ_1 times.

Step 2. **Split–merge step:**

1. For total of n observations, denote the whole set of observation as $U = \{1, \ldots, n\}$. Select two distinct observations i and j at random from U. Let A be a set of observations that have the same labels of i and j from U except i and j itself. In other words, A denotes the set of observations $l \in \{1, \ldots, n\}$, for which $l \neq i$, $l \neq j$, and $Z_l = Z_i$ or $Z_l = Z_j$.

2. If $Z_i = Z_j$, then set a launch state for Z_i^{launch} such that $Z_i^{launch} \notin \{Z_1, \ldots, Z_n\}$ and $Z_j^{launch} = Z_j$. Otherwise, $Z_i^{launch} = Z_i$ and $Z_j^{launch} = Z_j$. The launch state is the pre-Gibbs state which will turn into the reasonable proposal state by intermediate restricted Gibbs sampling scans.

3. For each $l \in A$, initialize Z_l^{launch} at random with probability 0.5 to either Z_i^{launch} or Z_j^{launch}.

4. Update Z_l^{launch} for $l \in A$ to either Z_i^{launch} or Z_j^{launch} by performing κ_2 intermediate restricted Gibbs sampling scans using the conditional distribution given by

$$p(Z_l | Z_{-l}, \mathbf{x}_l, \boldsymbol{\xi}) = \frac{n_{-l,Z_l} \int f(\mathbf{x}_l; \boldsymbol{\theta}, \boldsymbol{\xi}) dH_{-l,Z_l}(\boldsymbol{\theta}, \boldsymbol{\xi})}{n_{-l,Z_i} \int f(\mathbf{x}_l; \boldsymbol{\theta}, \boldsymbol{\xi}) dH_{-l,Z_i}(\boldsymbol{\theta}, \boldsymbol{\xi}) + n_{-l,Z_j} \int f(\mathbf{x}_l; \boldsymbol{\theta}, \boldsymbol{\xi}) dH_{-l,Z_j}(\boldsymbol{\theta}, \boldsymbol{\xi})}, \tag{8}$$

where Z_{-l} represents Z_r for $r \neq l$ in $A \cup \{i, j\}$; n_{-l,Z_l} is the number of Z_r terms for $r \neq l$ in $A \cup \{i, j\}$ that are equal to Z_l; H_{-l,Z_l} is the posterior distribution of $\boldsymbol{\theta}$ based on the prior G_0 and data observations $\mathbf{x}_r$ such that $Z_r = Z_l$, where $r \in A \cup \{i, j\}$ for

$r \neq l$; and

$$\int f(\mathbf{x}_l;\boldsymbol{\theta},\boldsymbol{\xi})dH_{-l,Z_i}(\boldsymbol{\theta},\boldsymbol{\xi}) = \pi^{-p_\xi/2}\left(\frac{h_1 n_{Z_i}+1}{h_1 n_{-l,Z_i}+1}\right)^{-p_\xi/2}\prod_{r=1}^{p_\xi}\frac{\Gamma\{(n_{Z_i}+\delta+p_\xi-r)/2\}}{\Gamma\{(n_{-l,Z_i}+\delta+p_\xi-r)/2\}}$$
$$\times\left|\boldsymbol{Q}_{1(\xi)}+\boldsymbol{S}_{Z_i(\xi)}\right|^{-(n_{Z_i}+\delta+p_\xi-1)/2}\left|\boldsymbol{Q}_{1(\xi)}+\boldsymbol{S}_{-l,Z_i(\xi)}\right|^{(n_{-l,Z_i}+\delta+p_\xi-1)/2}$$

with $\boldsymbol{S}_{g(\xi)}$ defined in Eq.(7) and

$$\boldsymbol{S}_{-l,Z_i(\xi)} = \sum_{r\neq l:Z_r=Z_i}(\mathbf{x}_{r(\xi)}-\overline{\mathbf{x}}_{Z_i(\xi)})(\mathbf{x}_{r(\xi)}-\overline{\mathbf{x}}_{Z_i(\xi)})^T+\frac{n_{-l,Z_i}}{h_1 n_{-l,Z_i}+1}(\boldsymbol{\mu}_{0(\xi)}-\overline{\mathbf{x}}_{Z_i(\xi)})(\boldsymbol{\mu}_{0(\xi)}-\overline{\mathbf{x}}_{Z_i(\xi)})^T.$$

5. If $Z_i = Z_j$, then propose a new split–assignment $Z_i^* = Z_i^{launch}$ and $Z_j^* = Z_j^{launch}$. The new assignment for an observation $l \in A$ is determined by performing one final Gibbs sampling scan from Z^{launch}, using Eq.(8) to set Z_l^* to either Z_i^* or Z_j^*. If $Z_i \neq Z_j$, then propose a new merge–assignment as $Z_i^* = Z_j$, $Z_j^* = Z_j$, and $Z_l^* = Z_j$ for $l \in A$. The allocation for observations $l \notin A \cup \{i,j\}$ remains unchanged for both cases. In both cases, the proposal is evaluated by the Metropolis–Hastings acceptance probability

$$a(\boldsymbol{Z}^*,\boldsymbol{Z}) = \min\left[1, \frac{q(\boldsymbol{Z}|\boldsymbol{Z}^*)}{q(\boldsymbol{Z}^*|\boldsymbol{Z})}\frac{p(\boldsymbol{Z}^*)}{p(\boldsymbol{Z})}\frac{L(\boldsymbol{Z}^*|\boldsymbol{X},\boldsymbol{\xi})}{L(\boldsymbol{Z}|\boldsymbol{X},\boldsymbol{\xi})}\right], \tag{9}$$

where $\boldsymbol{Z}$ and $\boldsymbol{Z}^*$ represent the original and proposal vectors, respectively; $q(\boldsymbol{Z}^*|\boldsymbol{Z})$ represents the Gibbs sampling transition probability from $\boldsymbol{Z}$ to $\boldsymbol{Z}^*$; $p(\boldsymbol{Z})$ represents the prior distribution of $\boldsymbol{Z}$; and $L(\boldsymbol{Z}|\boldsymbol{X},\boldsymbol{\xi})$ is the likelihood for $\boldsymbol{Z}$.

Step 3. **Full Gibbs sampling step:**

Update all sample allocations Z_i, $i = 1,\ldots,n$, from their conditional distributions

given by

$$
\begin{aligned}
&p(Z_i = Z_j \text{ for some } j \neq i | Z_{-i}, \mathbf{X}, \boldsymbol{\xi}) \\
\propto &\pi^{-np_\xi/2}(n_{-i,Z_j} + \alpha)\left(\frac{h_1 n_{Z_j} + 1}{h_1 n_{-i,Z_j} + 1}\right)^{-p_\xi/2} \prod_{r=1}^{p_\xi} \frac{\Gamma\{(n_{Z_j} + \delta + p_\xi - r)/2\}}{\Gamma\{(n_{-i,Z_j} + \delta + p_\xi - r)/2\}} \\
&\times \left|\boldsymbol{Q}_{1(\xi)} + \boldsymbol{S}_{Z_j(\xi)}\right|^{-(n_{Z_j}+\delta+p_\xi-1)/2} \left|\boldsymbol{Q}_{1(\xi)} + \boldsymbol{S}_{-i,Z_j(\xi)}\right|^{(n_{-i,Z_j}+\delta+p_\xi-1)/2},
\end{aligned} \tag{10}
$$

where $\boldsymbol{S}_{-i,Z_j(\xi)}$ is the equivalent expression without the ith observation, and

$$
\begin{aligned}
&p(Z_i \neq Z_j \text{ for all } j \neq i | Z_{-i}, \mathbf{X}, \boldsymbol{\xi}) \\
\propto &\pi^{-np_\xi/2} \frac{V_n(t+1)}{V_n(t)} \alpha (h_1 + 1)^{-p_\xi/2} \prod_{r=1}^{p_\xi} \frac{\Gamma\{(1 + \delta + p_\xi - r)/2\}}{\Gamma\{(\delta + p_\xi - r)/2\}} \\
&\times \left|\boldsymbol{Q}_{1(\xi)}\right|^{(\delta+p_\xi-1)/2} \left|\boldsymbol{Q}_{1(\xi)} + \boldsymbol{S}_{i(\xi)}\right|^{-(\delta+p_\xi)/2},
\end{aligned} \tag{11}
$$

where $\boldsymbol{S}_{i(\xi)} = (h_1 + 1)^{-1}(\mathbf{x}_{i(\xi)} - \mu_{0\xi})(\mathbf{x}_{i(\xi)} - \boldsymbol{\mu}_{0\xi})^T$.

The overall workflow of our model fitting procedure is summarized in Algorithm 1.

The above algorithm is as same as the model fitting procedure of Kim et al. (2006), except for the process of obtaining the prior distribution of latent variable $\boldsymbol{Z}$. $p(\boldsymbol{Z}) = \frac{\alpha^t}{\alpha^{(n)}} \prod_{c \in C}(|c| - 1)!$ which appears when calculating the Metropolis–Hasting acceptance probability of new assignments in Kim et al. (2006) was replaced by $p(\boldsymbol{Z}) = V_n(t) \prod_{c \in C} \alpha^{|c|}$ of Miller and Harrison (2018) where $|c|$ is the size of c and $V_n(t) = \sum_{k=1}^{\infty} \frac{k_{(t)}}{(k\alpha)^{(n)}} p_K(k)$. Eq.(17) and Eq.(18) of Kim et al. (2006) were replaced by the Eq.(10) and Eq.(11) of our model fitting procedure. We included all model fitting procedures including the previously proposed algorithm so that our paper is self-contained. Compare to DPM, MFM requires the calculation of $V_n(t)$ which seems a bit complicated. However, this does not have a significant impact on the overall calculation time since it only needs to be calculated once before the MCMC algorithm starts. Moreover, since $\frac{k_{(t)}}{(\alpha k)^{(n)}} \leq \frac{k^t}{(\alpha k)^{(n)}} \leq \frac{1}{(\alpha k)^{(n)}}$, series for $V_n(t)$ converges rapidly when $t \ll n$. This implies that $V_n(t)$ can easily be numerically approximated to a high level of precision.

Algorithm 1 Variable Selection in Clustering via MFM

Variable selection step

for iter = 1 to κ_1 **do**
  Generate a new candidate $\boldsymbol{\xi}^{new}$ with random transition move
  Determine whether to accept the new candidate by Metropolis-Hastings algorithm
  if $\boldsymbol{\xi}^{new}$ is accepted **then**
    Update $\boldsymbol{\xi}^{old} = \boldsymbol{\xi}^{new}$
  end if
end for

Split–merge step

Choose two distinct observation i and j
Set A the set of observations $l \in \{1, \ldots, n\}$, where $l \neq i$, $l \neq j$, and $Z_l = Z_i$ or $Z_l = Z_j$
if $Z_i = Z_j$ **then**
  Set Z_i^{launch} such that $Z_i^{launch} \notin \{Z_1, \ldots, Z_n\}$ and $Z_j^{launch} = Z_j$
else
  $Z_i^{launch} = Z_i$ and $Z_j^{launch} = Z_j$
end if
for iter = 1 to κ_2 **do**
  Update Z_l^{launch} for $l \in A$ to either Z_i^{launch} or Z_j^{launch} by performing κ_2 intermediate restricted Gibbs sampling scans
end for
if $Z_i = Z_j$ **then**
  Let $Z_i^* = Z_i^{launch}$ and $Z_j^* = Z_j^{launch}$
else
  Let $Z_i^* = Z_j$ and $Z_j^* = Z_j$
  For every observation $l \in A$, let $Z_l^* = Z_j$
end if
Determine whether to accept the new assignment, $\boldsymbol{Z}^*$, by Metropolis-Hastings algorithm
if $\boldsymbol{Z}^*$ is accepted **then**
  Update $\boldsymbol{Z} = \boldsymbol{Z}^*$
end if

Full Gibbs sampling step

for i = 1 to n **do**
  Update sample allocations Z_i, $i = 1, \ldots, n$, from their conditional distributions
end for

3.2 Posterior Inference

Since our model fitting procedure is made up of Metropolis–Hasting algorithm of variable selection step and split–merge algorithm of clustering step, posterior inference has been done separately in each MCMC algorithm. For the inference of the discriminating variables, we employ the largest marginal posterior probabilities $p(\xi_j = 1|\boldsymbol{X})$, which are estimated by the empirical frequencies in the MCMC outputs. Next, we utilize the maximum a posteriori sample allocation vector to infer the cluster structure, as follows:

$$\hat{\boldsymbol{Z}} = \underset{1 \leqslant m \leqslant M}{\operatorname{argmax}}\ p(\boldsymbol{Z}^{(m)}|\boldsymbol{X}, \hat{\boldsymbol{\xi}}), \tag{12}$$

where $\boldsymbol{Z}^{(m)}$ is the cluster allocation vector of the mth MCMC sample and $\hat{\boldsymbol{\xi}}$ is the set of selected variables based on the marginal posterior probabilities. If there is a burn-in period, m should start after burning phase. We also calculated the proportion of iterations that any given pair of samples cluster together for estimating the posterior pairwise probability, $p(Z_i = Z_j|\boldsymbol{X})$, of clustering. Next, we made symmetric $n \times n$ similarity matrix with $\binom{n}{2}$ pairwise posterior probabilities.

4 Experiments

4.1 Simulated Example

As we mentioned in Section 2.1, MFM model has the advantage over DPM model that it produces fewer tiny clusters and shows consistent results when the number of observations increases. To verify this, we conducted experiments under the same condition as Kim et al. (2006) which used the DPM model. We also conducted additional experiments under different conditions where data is sparsely distributed to see if MFM model still produces fewer tiny clusters even for the complex datasets. Finally, in order to verify that the number

of clusters is consistently estimated even when the value of the observation increases, the experiment was carried out by increasing the number of data.

We evaluate the performance of our model using simulated data. For comparison of our MFM model with the DPM counterpart of Kim et al. (2006), we generated data in the same manner as in Kim et al. (2006). A dataset of 15 observations and 1000 variables was assumed. Among the 1000 variables, 20 were assumed to discriminate the observations into four clusters, as follows:

$$x_{i,j} \sim I_{\{1\leqslant i\leqslant 4\}}N(\mu_1,\sigma_1^2) + I_{\{5\leqslant i\leqslant 7\}}N(\mu_2,\sigma_2^2) + I_{\{8\leqslant i\leqslant 13\}}N(\mu_3,\sigma_3^2) + I_{\{14\leqslant i\leqslant 15\}}N(\mu_4,\sigma_4^2), \quad (13)$$

where i is the observation index $i = 1,\ldots,15$; j is the variable index $j = 1,\ldots,20$; and $I_{\{\cdot\}}$ is the indicator function. The component parameters μ_g and σ_g^2 for $g = 1,\ldots,4$ were randomly chosen from $[-5,5]$ and $[0.01,1]$, respectively. The remaining 980 nondiscriminating variables were generated from a standard normal distribution. We conducted experiments several times with different datasets from the generating process and compared the overall result with the result of Kim et al. (2006). For the accurate comparison, we also specified a single dataset from the generating process and used it for comparing the results of the two approaches.

The hyperparameters were set as follow: $h_1 = 1000$, $h_0 = 100$, $k_1 = 2$, $\delta = 3$, $a = 3$, $b = 2$, and $\omega = 0.01$. In addition, the concentration parameter α was set to 1. In fact, we repeated the experiments with various α values while keeping the other parameter values unchanged, but obtained reasonably robust results.

We initialized $\boldsymbol{\xi}$ by setting its one randomly chosen element to 1 and others to 0. For the initial iteration, we assigned each sample to a different cluster by setting the Z_i terms to have different values. We ran 100,000 iterations, including the first 40,000 samples for a burn-in period. As we mentioned above, κ_1 is the number of repeating Metropolis step of variable selection. If the total number of iterations is large enough and the MCMC converges

to a stationary state, the latent binary vector, $\boldsymbol{\xi}$, will also converge to the same value even though κ_1 is small. While convergence rate of $\boldsymbol{\xi}$ and overall iteration time shows in the trade-off relationship depending on κ_1, we found out that setting $\kappa_1 = 20$ works best in our experiments. Table 1 shows how convergence rate of $\boldsymbol{\xi}$ and iteration time change depending on κ_1. κ_2 is the number of repeating intermediate restricted Gibbs sampling scans. A large number of restricted scans makes the split proposal more reasonable and therefore more likely to be accepted. However, it takes longer computation time and benefit from the reasonable proposal reaches the limit at some moment (Dahl, 2003). We set κ_2 to 5, following the recommendation of Jain and Neal (2004) that the improvement in mixing is minimal after five intermediate scans.

Table 1: Effects of κ_1 on convergence rate of $\boldsymbol{\xi}$ and iteration time

κ_1	**No. of iterations until convergence**	**computation time per iteration**
1	364	0.2036 seconds
10	41	0.6252 seconds
20	19	1.1600 seconds
30	14	1.8084 seconds

Table 2: Comparison of MFM and DPM models for the simulation example (true number of clusters: 4, true number of discriminating variables: 20)

		No. of clusters	**No. of selected variables**	**ARI**
MFM	$\alpha = 1$	4	20	1
	$\alpha = 15$	4	20	1
DPM	$\alpha = 1$	5	17	0.9734
	$\alpha = 15$	5	15	0.9734

Our results are summarized in Table 2, together with the results of using DPMs instead of MFMs reported in Kim et al. (2006). We also used Adjusted Rand Index (ARI) of Hubert and Arabie (1985) to show the performance difference between the two models. In most datasets, our model successfully identified 4 clusters without generating one-sample clusters.

Table 3: Comparison of MFM and DPM models for different simulation data

			MFM			DPM		
			No. of clusters	**No. of selected variables**	**ARI**	**No. of clusters**	**No. of selected variables**	**ARI**
$n = 15$	$\sigma = 0.5$	$\alpha = 1$	4	20	1	4	23	1
		$\alpha = 15$	4	20	1	4	24	1
	$\sigma = 2$	$\alpha = 1$	4	18	1	4	25	1
		$\alpha = 15$	4	17	1	5	27	0.9734
$n = 30$	$\sigma = 0.5$	$\alpha = 1$	4	20	1	4	20	1
		$\alpha = 15$	4	20	1	5	22	0.9734
	$\sigma = 2$	$\alpha = 1$	4	19	1	7	20	0.7640
		$\alpha = 30$	4	19	1	8	21	0.6598

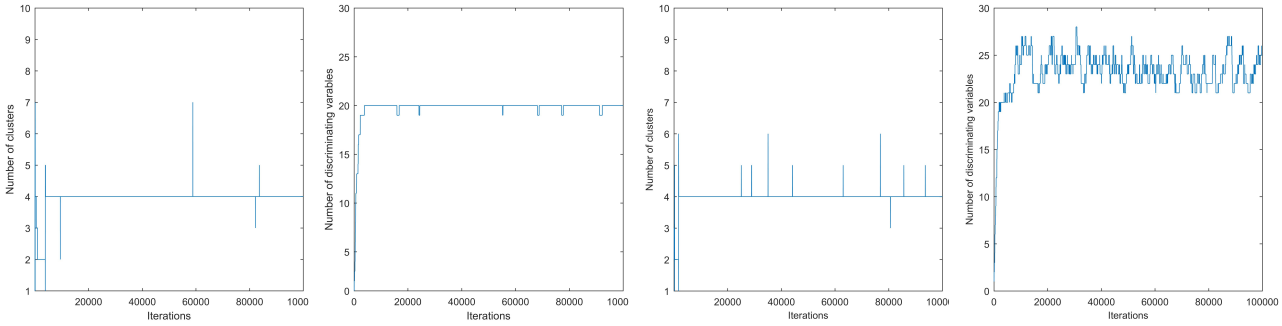
We also observe the results were robust to changes in the concentration parameter. When we set α to 15, which is equal to the sample size, the numbers of clusters and discriminating variables were the same as for the case with α=1. On the other hand, the experiment of Kim et al. (2006) estimated five components with the last two observations assigned to separate clusters. While our model estimated the numbers of clusters and discriminating variables accurately for both values of α, the DPM model overestimated the number of clusters and underestimated the number of discriminating variables for both values of α. It is also observed the DPM model was less robust than ours, producing different results according to the α values.

For the accurate comparison of two experiments with same dataset, we generated a dataset with the same way of Eq.(13) but fixed component parameters as follow: $\{(\mu_1, \sigma_1) = (-4, 0.5), (\mu_2, \sigma_2) = (-1, 0.5), (\mu_3, \sigma_3) = (2, 0.5), (\mu_4, \sigma_4) = (5, 0.5)\}$ and $\{(\mu_1, \sigma_1) = (-4, 2), (\mu_2, \sigma_2) = (-1, 2), (\mu_3, \sigma_3) = (2, 2), (\mu_4, \sigma_4) = (5, 2)\}$. The first four rows of Table 3 present the result of the experiments. For both datasets, both MFM and DPM model successfully identified 4 clusters except DPM when $\alpha = 15$ and $\sigma = 2$. On the other hand, MFM and DPM showed different results from estimating the numbers of discriminating variables. When all values of σ were set to 0.5, MFM model correctly identified 20 discriminating variables, while DPM model did not correctly identified discriminating variables and overestimated the

number of discriminating variables. When all values of σ were set to 2, both MFM and DPM did not estimated the correct number of discriminating variables: MFM identified 18 and 17 discriminating variables, respectively, when α was set to 1 and 15 and DPM identified 25 and 27 discriminating variables, respectively. Figures 1 and Figures 2 present the trace plots for the estimated numbers of clusters and discriminating variables when $\alpha = 1$. When all values of σ were set to 0.5, it is clear that our sampler stabilized to the true model with four clusters and 20 discriminating variables. However, under DPM model, the chain did not stabilized and oscillated around 23. When all values of σ were set to 2, both MFM and DPM model did not correctly identified discriminating variables and corresponding chains were not stabilized. Nevertheless, it is clear that the chains of our model is more close to the value true value with less fluctuation compare to the chain of DPM model.

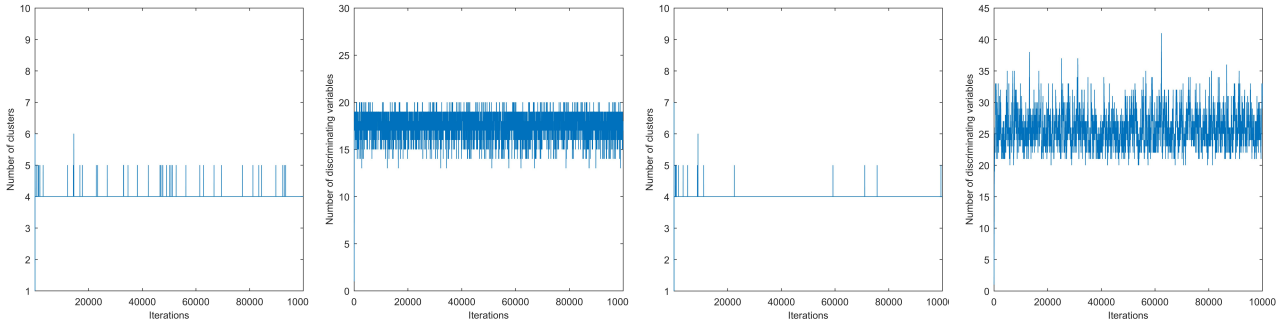
We also generated another dataset with the increased number of observations. We maintained the cluster structure by doubling the observations belonging to each cluster. With $n = 30$, we set all values of σ to 2. The result is on the last two rows of Table 3. While our model successfully identified 4 clusters, DPM overestimated the number of clusters to 7 and 8. This can be interpreted that DPM tend to create many tiny extra clusters with more samples, while MFM does not. Figures 3 presents the trace plots for the estimated numbers of clusters and discriminating variables when $\alpha = 1$. It is clear that our sampler stabilized to the true model with four clusters, but under DPM model, the chain did not stabilized and oscillated from 5 to 9.

Furthermore, to show that our model estimate the number of clusters consistently even when the number of observations increases, we repeated the experiments by generating new datasets, by increasing the number of observations as 30, 60, and 120 while keeping the cluster structure the same as in the original experiments with 15 observations. Here, we fixed component parameters as follow: $\{(\mu_1, \sigma_1) = (-4, 0.5), (\mu_2, \sigma_2) = (-1, 0.5), (\mu_3, \sigma_3) = (2, 0.5), (\mu_4, \sigma_4) = (5, 0.5)\}$ when generating datasets. We kept the cluster structure the



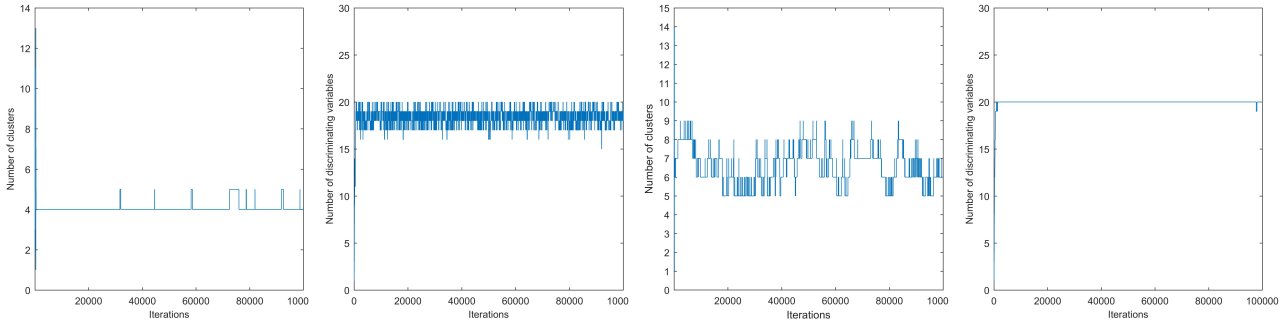
(a) Number of clusters (MFM) (b) Number of included discriminating variables (MFM) (c) Number of clusters (DPM) (d) Number of included discriminating variables (DPM)

Figure 1: Trace plots for the simulated example ($\alpha = 1$, $\sigma_1 = \sigma_2 = \sigma_3 = \sigma_4 = 0.5$, $n = 15$)



(a) Number of clusters (MFM) (b) Number of included discriminating variables (MFM) (c) Number of clusters (DPM) (d) Number of included discriminating variables (DPM)

Figure 2: Trace plots for the simulated example ($\alpha = 1$, $\sigma_1 = \sigma_2 = \sigma_3 = \sigma_4 = 2$, $n = 15$)



(a) Number of clusters (MFM) (b) Number of included discriminating variables (MFM) (c) Number of clusters (DPM) (d) Number of included discriminating variables (DPM)

Figure 3: Trace plots for the simulated example ($\alpha = 1$, $\sigma_1 = \sigma_2 = \sigma_3 = \sigma_4 = 2$, $n = 30$)

same as in the original experiments with 15 observations, i.e., assuming four components. For example, a dataset of 30 observations was generated as follows:

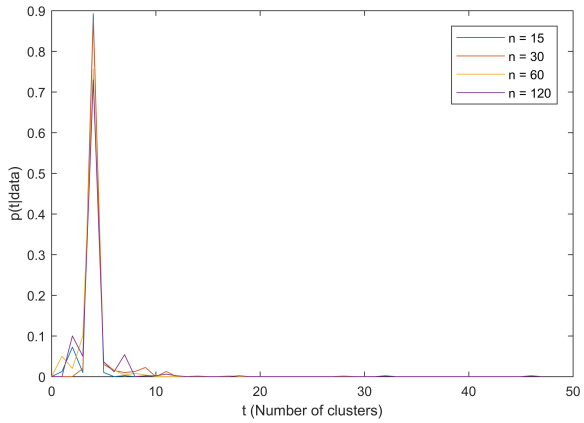
$$x_{i,j} \sim I_{\{1 \leqslant i \leqslant 8\}} N(\mu_1, \sigma_1^2) + I_{\{9 \leqslant i \leqslant 14\}} N(\mu_2, \sigma_2^2) + I_{\{15 \leqslant i \leqslant 26\}} N(\mu_3, \sigma_3^2) + I_{\{27 \leqslant i \leqslant 30\}} N(\mu_4, \sigma_4^2)$$

with the same component parameters μ_g and σ_g^2, $g = 1, \ldots, 4$ as used in Eq.(13). Here, α was set to 1 in both MFM and DPM.

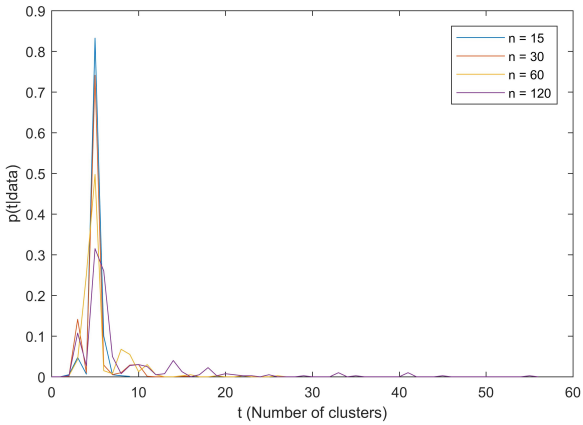
In this experiment, both MFM and DPM successfully identified 4 clusters for all data. While 20 discriminating variables were identified in MFM for all datasets, DPM overestimated the number of discriminating variables to 22 and 23 for $n = 60$ and $n = 120$ respectively. Figures 4(a) and 4(b) depict the posterior distributions on the number of clusters (denoted by t) obtained using the MFM and DPM models, respectively. Even though both MFM and DPM successfully identified 4 clusters for all data, but posterior distributions of two models are significantly different: distributions of DPM is more skewed to the right than MFMs'. When the number of observations was small, at 15 or 30, both the MFM and DPM models tended to generate few tiny clusters. However, as the number of observations increased the DPM model tended to place higher probabilities on partitions with tiny clusters, while the posterior probability for the MFM model concentrated around the true value of the number of clusters.

To summarize the results of the simulation studies, the proposed model showed better performance than the existing model for all datasets. In specific, our model produced fewer tiny clusters and showed consistent results with the different number of observations. This implies that our model is appropriate for the dataset with the limited number of clusters, regardless of the number of observations. For example, clustering patients according to the microarray dataset is appropriate to our model because the types of genetic disorder expressed is limited even if the number of patients increases.

As we mentioned in section 3.1, the computational difference from DPM model when



(a) Variable selection with MFMs



(b) Variable selection with DPMs

Figure 4: Posterior on the number of clusters t for the experiments using DPM and MFM

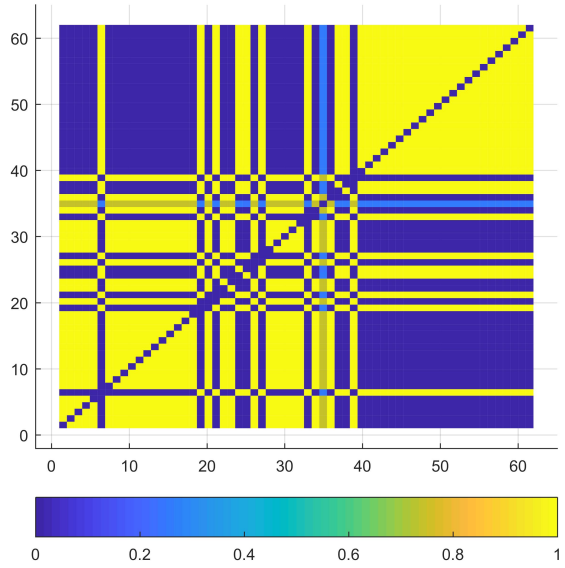
inferring the MFM model is that $V_n(t) = \sum_{k=1}^{\infty} \frac{k_{(t)}}{(k\alpha)^{(n)}} p_K(k)$ must be calculated for all possible t values before the MCMC algorithm starts. In our simulation studies, it took about 0.1 seconds to calculate the value of $V_n(t)$, and took 10 seconds to calculate all values of $V_n(t)$ for the possible number of clusters, t. Since the number of clusters cannot exceeds the number of observations, calculation of $V_n(t)$ does not have a significant impact on the overall calculation time when there aren't many observations.

4.2 Colon Cancer Data Example

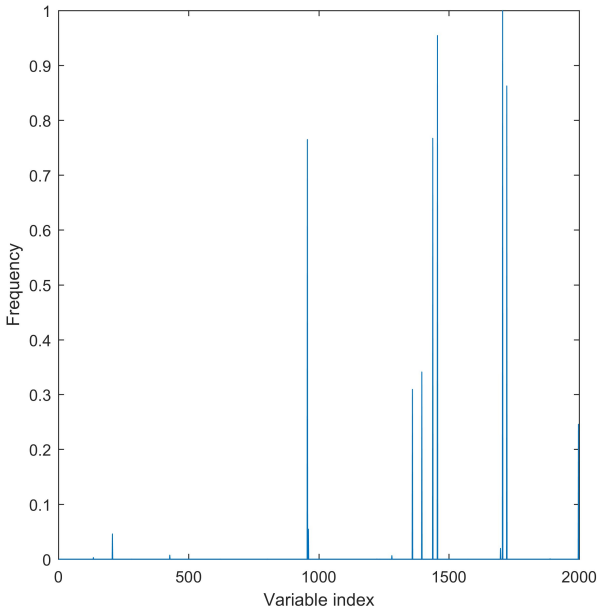
We used a real dataset of Affymetrix oligonucleotide array measurements of gene expression levels for 40 tumor and 22 normal colon tissue samples for 6500 human genes (Alon et al., 1999). For the analysis, we employed 2000 selected genes with the highest minimal intensities across the samples. Thus, the microarray data matrix had 2000 columns and 62 rows. We arranged the rows such that tumors are labeled by the row numbers 1 through 40 and the normals are labeled by the row numbers 41 through 62.

For comparison, we conducted experiments using the MFM and DPM models under the same conditions. The hyperparameters were set as $h_1 = 10$, $h_0 = 100$, $k_1 = 3$, $\delta = 0.1$, $a = 0.1$, $b = 7$, and $\omega = 0.03$. The concentration parameter was set as $\alpha = 1$ for both MFM

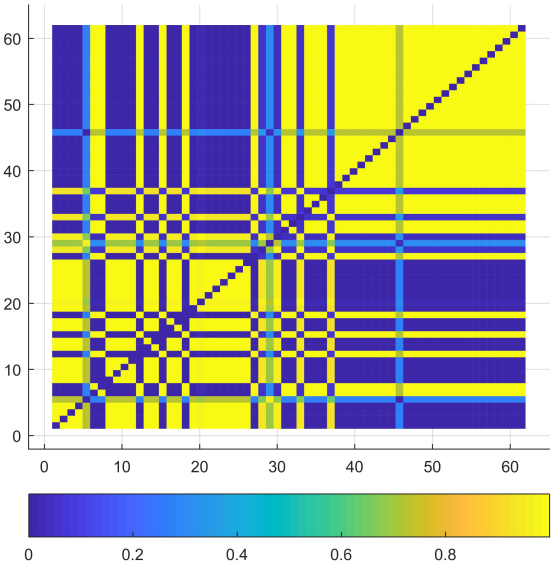
model and the DPM model. For both models, we ran 100,000 iterations, including the first 40,000 samples discarded for burn-in. At the initial iteration, all samples were assigned to the same cluster, and only one randomly chosen ξ_j was set to 1. For the algorithm in Section 3.1, we set $\kappa_1 = 20$ for *Step 1* and $\kappa_2 = 3$ for *Step 2*.



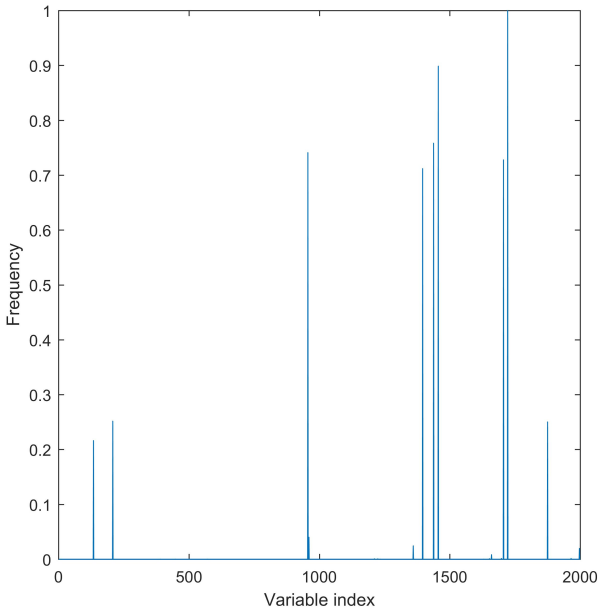
(a) Pairwise posterior probabilities $p(Z_i = Z_j|\mathbf{X})$ (MFM)



(b) Marginal posterior probabilities $p(\xi_i = 1|\mathbf{X})$ (MFM)



(c) Pairwise posterior probabilities $p(Z_i = Z_j|\mathbf{X})$ (DPM)



(d) Marginal posterior probabilities $p(\xi_i = 1|\mathbf{X})$ (DPM)

Figure 5: Colon cancer data of 2000 genes (40 tumor and 22 normal colon tissue samples)

Figure 5 summarizes the results of the MFM and DPM models. Figures 5(a) and 5(c)

display heatmaps of the pairwise posterior probabilities $p(Z_i = Z_j|\mathbf{X})$ using the MFM and DPM models, respectively. Pairwise posterior probabilities were estimated from the empirical frequencies in the Markov chain Monte Carlo output. Since we counted how often two different observations, not the one observation itself, belong to the same cluster for the whole MCMC output, the values of 0 were assigned to the diagonal even though one sample has a probability of 1 to be assigned to the same cluster. The first 40 indices correspond to the tumor samples, and the last 22 correspond to the normal samples. If $p(Z_i = Z_j|\mathbf{X})$ is close to 1 for the variables i and j, this can be interpreted as implying that i and j belong to the same cluster with a high probability. We can observe that the first 40 and last 22 indices are more distinctly divided in Figure 5(a) for the MFM model compared to Figure 5(c) for the DPM model. The detailed cluster structure inferred from the maximum a posteriori sample allocation vector is described in Table 5. Both the MFM and DPM models assigned all normal colon tissues correctly to the same cluster. However, both models assigned some of the tumor colon tissues to the cluster of normal tissues. Of the two models, the MFM model resulted in fewer misclustered tumor tissues than the DPM model (9 versus 14). Along with the Adjusted Rand Index, we added three performance measures which have been widely used to evaluate the performance of the clustering algorithms to clarify the result: Rand Index (Rand, 1971), F-score (Achtert et al., 2012) and V-score (Rosenberg and Hirschberg, 2007). Table 4 shows that MFM model outperformed DPM model with all performance measures.

Table 4: Comparison of MFM and DPM models for Colon cancer data

	MFM	**DPM**
Rand index	0.7478	0.6446
Adjusted Rand Index	0.4961	0.2888
F-score	0.7543	0.6582
V-score	0.5198	0.3944

The marginal posterior probabilities of the latent binary vector $\boldsymbol{\xi}$ are shown in Fig-

ures 5(b) and 5(d) for the MFM and DPM models, respectively. Using the MFM model, five genes were selected as discriminating variables with marginal posterior probabilities greater than 0.7, while one additional gene was selected using the DPM model. The selected discriminating gene variables are summarized in Table 6. Among the selected genes, $\{Z15115, H71627, H09273\}$ were also identified as related to colon cancer in other studies (Archetti et al., 2010; Li and Li, 2008; Huang and Kecman, 2005).

Table 5: The estimated cluster structures for the colon cancer data using MFM and DPM

MFM	**Tumor (40)**	**Normal (22)**	**DPM**	**Tumor (40)**	**Normal (22)**
Cluster l	31	0	Cluster l	26	0
Cluster 2	9	22	Cluster 2	14	22

Table 6: The selected discriminating genes for the colon cancer data example

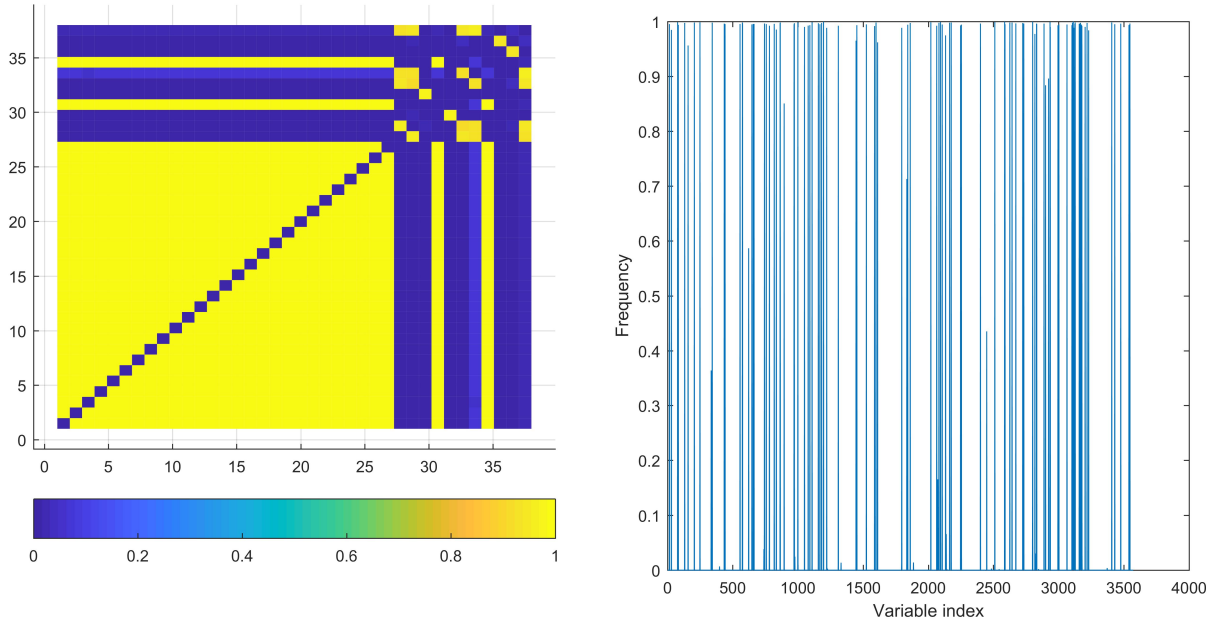
	Sample index	**Clone ID**	**Gene annotation**
DPM	956	X03484	Human mRNA for raf oncogene
	1396	H42127	yo61b11.s1 Soares breast 3NbHBst Homo sapiens cDNA clone
	1438	D26069	Homo sapiens KIAA0041 mRNA
	1456	Z15115	H.sapiens TOP2 mRNA for DNA topoisomerase II (partial)
	1706	H71627	VITELLOGENIN A2 PRECURSOR (Xenopus laevis)
	1722	H09273	Putative 118.2 kd transcriptional regulatory protein in ACS1-PTA1 intergenic region (Saccharomyces cerevisiae)
MFM	956	X03484	Human mRNA for raf oncogene
	1438	D26069	Homo sapiens KIAA0041 mRNA
	1456	Z15115	H.sapiens TOP2 mRNA for DNA topoisomerase II (partial)
	1706	H71627	VITELLOGENIN A2 PRECURSOR (Xenopus laevis)
	1722	H09273	Putative 118.2 kd transcriptional regulatory protein in ACS1-PTA1 intergenic region (Saccharomyces cerevisiae)

4.3 Leukemia Data Example

We further applied our model to a real dataset of leukemia tissue samples from Golub et al. (1999), where the authors studied gene expressions on two types of acute leukemia: acute lymphoblastic leukemia (ALL) and acute myeloid leukemia (AML). We focus on the 38 patients from the training set of the experiments of Golub et al. (1999), consisting of 27 ALL

and 11 AML patients. We followed the same data preprocessing steps as in Dudoit et al. (2002). First, we truncated expression measures beyond the thresholds of reliable detection at 100 and 16,000. Next, we excluded genes with max/min $\leqslant 5$ and (max - min) $\leqslant 500$, where max and min refer to the maximum and minimum expression levels of a particular gene across samples, respectively. This filtering resulted in 3571 genes. Finally, the expression measures were log-transformed and each variable was rescaled by its range.

We applied the MFM model to the data and compared the results with those of the DPM model reported in Kim et al. (2006). For a fair comparison, the hyperparameters were set the same as in Kim et al. (2006): $h_1 = 10$, $h_0 = 100$, $k_1 = 0.06$, $\delta = 3$, $a = 3$, $b = 0.1$, and $\omega = 0.005$. In addition, the concentration parameter α was set to 1. For the algorithm in Section 3.1, we set $\kappa_1 = 20$ for *Step 1* and $\kappa_2 = 3$ for *Step 2*.



(a) Pairwise posterior probabilities $p(Z_i = Z_j|\mathbf{X})$ (MFM)

(b) Marginal posterior probabilities $p(\xi_i = 1|\mathbf{X})$ (MFM)

Figure 6: Leukemia data of 3571 genes (27 ALL and 11 AML samples)

Figure 6(a) presents a heatmap of the pairwise posterior probabilities. We can observe that all of the first 27 indices of the ALL samples were allocated to the same cluster with high

probabilities. On the other hand, the last 11 AML samples exhibited less homogeneity. The sample allocation estimates based on the maximum a posteriori probability were represented in Table 7.

Table 7: The estimated cluster structures for the leukemia data using MFM and DPM

	MFM	**ALL (27)**	**AML (11)**	**DPM**	**ALL (27)**	**AML (11)**
$\alpha = 1$	Cluster 1	27	2	Cluster 1	25	0
	Cluster 2	0	6	Cluster 2	2	5
	Cluster 3	0	2	Cluster 3	0	2
	Cluster 4	0	1	Cluster 4	0	1
				Cluster 5	0	2
				Cluster 6	0	1
$\alpha = 38$	Cluster 1	26	4	Cluster 1	24	1
	Cluster 2	1	4	Cluster 2	3	4
	Cluster 3	0	2	Cluster 3	0	2
	Cluster 4	0	1	Cluster 4	0	1
				Cluster 5	0	1
				Cluster 6	0	1
				Cluster 7	0	1

It is clear that the ALL and AML samples were successfully separated, even though the AML samples were separated into four groups including the ALL group. For comparison, for the DPM model the AML samples were allocated to a total of six clusters, and even the ALL samples were allocated to two clusters (Kim et al., 2006). The DPM model also generated more tiny clusters than the MFM model. While, MFM generated only one singleton cluster, DPM generated four singleton clusters. The results with four performance measures are represented in Table 8. Figure 6(b) shows the marginal posterior probabilities $p(\xi_j = 1|\mathbf{X})$, $j = 1, \ldots, 3571$. With a marginal probability threshold 0.7, our model selected 109 genes as discriminating variables, while the DPM model selected 112 variables (Kim et al., 2006).

Table 8: Comparison of MFM and DPM models for Leukemia data

	MFM	**DPM**
Rand index	0.8691	0.7781
Adjusted Rand Index	0.7299	0.5619
F-score	0.8889	0.7857
V-score	0.6076	0.5309

For a sensitivity analysis of our model, we repeated the experiments with a new value of α. We set $\alpha = 38$ as Kim et al. (2006) conducted experiments with $\alpha = 1$ and $\alpha = 38$. With $\alpha = 38$, 138 genes were chosen as discriminating variables, and the sample allocation estimates were represented in Table 7. The ALL and AML samples were unclearly separated compared to the results for $\alpha = 1$. However, the number of generated clusters remained the same as four, while the number of clusters for the DPM model with $\alpha = 38$ was increased to seven according to Kim et al. (2006). The third and fifth columns of Table 7 summarizes the numbers of clusters estimated using the MFM and DPM models.

As we can see from the MFM representation in the Section 1, prior information about the relative sizes of the component weights $(\pi_1, \ldots, \pi_k)$ can be introduced through α. In other words, small α favors lower entropy $\boldsymbol{\pi}$, and large α favors higher entropy $\boldsymbol{\pi}$. The value of α must be chooen with a prior on the number of components: we recommend to set $\alpha = 1$ when small number of clusters is expected and set α equal to the number of observations when large number of clusters is expected. Without any prior on the number of components, Richardson and Green (1997) and Miller and Harrison (2018) recommended to set $\alpha = 1$ with empirical studies with many datasets.

5 Conclusion

We proposed a new method to simultaneously perform clustering and variable selection for high-dimensional data based on MFMs. Compared to DPM models, MFM models tend to

place small probabilities on partitions with tiny clusters, and enable the consistent estimation of the number of clusters. Using simulated and real data examples, we demonstrated the effective performance of the proposed method in terms of both clustering and variable selection.

Our assumption that variables relavent in clustering are independent from the irrelevant variables was appropriate to the DNA datasets with some previous researches that showed microarray data are not always correlated themselves (Zhao et al., 2007; Hirakawa et al., 2011). However, there are still many types of high–dimensional data in the real world whose all variables are corelated or the correlation between variables is unknown. Therefore, there could be an extension of this research to deal with entirely correlated dataset. Similar attempts have been done by Raftery and Dean (2006) who used Bayesian information criterion to accomplish variable selection with a greedy search, and Celeux et al. (2019) who suggested hybrid approach where a LASSO-like procedure and the model selection algorithm are employed together.

Since the relationship between genes and diseases has not been fully elucidated, there is no evidence showing that tumor/normal or AML/ALL have to be homogeneous respect to gene expression. This implies that DPM could have found the heterogeneity among patients and using MFM which allocates small probabilities to tiny clusters is not always better than using DPM for DNA microarray data. Nevertheless, we showed that MFM estimate the number of clusters consistently with changes in the number of observations with simulation studies. Thus, by increasing the number of observations, it would be possible to verify the homogeneity among patients. If patients are homogenous, MFM will consistently estimate small number of clusters while DPM will generate more clusters with more observations. Otherwise, both MFM and DPM will estimate more clusters with more observations.

In addition, we assumed the same set of discriminating variables across all clusters in this study. In future research, it may be possible to extend the proposed model by considering

separate sets of variables for different clusters, similar to Hoff et al. (2006).

References

Achtert, E., Goldhofer, S., Kriegel, H.-P., Schubert, E., and Zimek, A. (2012), "Evaluation of clusterings–metrics and visual support," in *2012 IEEE 28th International Conference on Data Engineering*, IEEE, pp. 1285–1288.

Alon, U., Barkai, N., Notterman, D. A., Gish, K., Ybarra, S., Mack, D., and Levine, A. J. (1999), "Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays," *Proceedings of the National Academy of Sciences*, 96, 6745–6750.

Archetti, F., Castelli, M., Giordani, I., and Vanneschi, L. (2010), "Classification of colon tumor tissues using genetic programming," in *Artificial Life and Evolutionary Computation*, World Scientific, pp. 49–58.

Blackwell, D., MacQueen, J. B., et al. (1973), "Ferguson distributions via Pólya urn schemes," *The Annals of Statistics*, 1, 353–355.

Celeux, G., Maugis-Rabusseau, C., and Sedki, M. (2019), "Variable selection in model-based clustering and discriminant analysis with a regularization approach," *Advances in Data Analysis and Classification*, 13, 259–278.

Dahl, D. B. (2003), "An improved merge-split sampler for conjugate Dirichlet process mixture models," *Technical R eport*, 1, 086.

Dudoit, S., Fridlyand, J., and Speed, T. P. (2002), "Comparison of discrimination methods for the classification of tumors using gene expression data," *Journal of the American Statistical Association*, 97, 77–87.

Friedman, J. H. and Meulman, J. J. (2004), "Clustering objects on subsets of attributes (with discussion)," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 66, 815–849.

Geng, J., Bhattacharya, A., and Pati, D. (2018), "Probabilistic community detection with unknown number of communities," *Journal of the American Statistical Association*, accepted, doi:10.1080/01621459.2018.1458618.

Golub, T. R., Slonim, D. K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J. P., Coller, H., Loh, M. L., Downing, J. R., Caligiuri, M. A., et al. (1999), "Molecular classification of cancer: class discovery and class prediction by gene expression monitoring," *Science*, 286, 531–537.

Grazian, C., Villa, C., and Liseo, B. (2020), "On a loss-based prior for the number of components in mixture models," *Statistics & Probability Letters*, 158, 108656.

Hirakawa, A., Hamada, C., and Yoshimura, I. (2011), "Sample size calculation for a regularized t-statistic in microarray experiments," *Statistics & probability letters*, 81, 870–875.

Hoff, P. D. et al. (2006), "Model-based subspace clustering," *Bayesian Analysis*, 1, 321–344.

Huang, T. M. and Kecman, V. (2005), "Gene extraction for cancer diagnosis by support vector machinesan improvement," *Artificial Intelligence in Medicine*, 35, 185–194.

Hubert, L. and Arabie, P. (1985), "Comparing partitions," *Journal of classification*, 2, 193–218.

Jain, S. and Neal, R. M. (2004), "A split-merge Markov chain Monte Carlo procedure for the Dirichlet process mixture model," *Journal of computational and Graphical Statistics*, 13, 158–182.

Kim, S., Tadesse, M. G., and Vannucci, M. (2006), "Variable selection in clustering via Dirichlet process mixture models," *Biometrika*, 93, 877–893.

Li, S. and Li, D. (2008), *DNA microarray technology and data analysis in cancer research*, Singapore: World Scientific.

Malsiner-Walli, G., Frühwirth-Schnatter, S., and Grün, B. (2016), "Model-based clustering based on sparse finite Gaussian mixtures," *Statistics and computing*, 26, 303–324.

Maugis, C., Celeux, G., and Martin-Magniette, M.-L. (2009), "Variable selection for clustering with Gaussian mixture models," *Biometrics*, 65, 701–709.

Miller, J. W. and Harrison, M. T. (2013), "A simple example of Dirichlet process mixture inconsistency for the number of components," in *Advances in Neural Information Processing Systems*, vol. 26, pp. 199–206.

— (2018), "Mixture models with a prior on the number of components," *Journal of the American Statistical Association*, 113, 340–356.

Murphy, K. P. (2012), *Machine learning: a probabilistic perspective*, MIT press.

Nobile, A. and Fearnside, A. T. (2007), "Bayesian finite mixtures with an unknown number of components: The allocation sampler," *Statistics and Computing*, 17, 147–162.

Raftery, A. E. and Dean, N. (2006), "Variable selection for model-based clustering," *Journal of the American Statistical Association*, 101, 168–178.

Rand, W. M. (1971), "Objective criteria for the evaluation of clustering methods," *Journal of the American Statistical association*, 66, 846–850.

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, 731–792.

Rosenberg, A. and Hirschberg, J. (2007), "V-measure: A conditional entropy-based external cluster evaluation measure," in *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)*, pp. 410–420.

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

Wasserman, L. (2000), "Asymptotic inference for mixture models by using data-dependent priors," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 62, 159–180.

Yau, C. and Holmes, C. (2011), "Hierarchical Bayesian nonparametric mixture models for clustering with variable relevance determination," *Bayesian analysis (Online)*, 6, 329.

Zhao, Y., Yu, J. X., Wang, G., Chen, L., Wang, B., and Yu, G. (2007), "Maximal subspace coregulated gene clustering," *IEEE Transactions on Knowledge and Data Engineering*, 20, 83–98.