Bayesian Analysis (2009) 4, Number 2, pp. 243–264

Modal Clustering in a Class of Product Partition Models

David B. Dahl*

Abstract. This paper defines a class of univariate product partition models for which a novel deterministic search algorithm is guaranteed to find the maximum *a posteriori* (MAP) clustering or the maximum likelihood (ML) clustering. While the number of possible clusterings of n items grows exponentially according to the Bell number, the proposed mode-finding algorithm exploits properties of the model to provide a search requiring only $n(n+1)$ computations. No Monte Carlo is involved. Thus, the algorithm finds the MAP or ML clustering for potentially tens of thousands of items, whereas it can only be approximated through a stochastic search. Integrating over the model parameters in a Dirichlet process mixture (DPM) model leads to a product partition model. A simulation study explores the quality of the clustering estimates despite departures from the assumptions. Finally, applications to three specific models — clustering means, probabilities, and variances — are used to illustrate the variety of applicable models and mode-finding algorithm.

Keywords: Bayesian nonparametrics, Dirichlet process mixture model, model-based clustering, maximum a posteriori clustering, maximum likelihood clustering, product partition models

1 Introduction

This paper considers a class of univariate product partition models (Hartigan 1990; Barry and Hartigan 1992) whose properties are such that a proposed algorithm is guaranteed to find the global maximum *a posteriori* (MAP) clustering of the posterior clustering distribution. The MAP clustering is the clustering estimate that minimizes the posterior expectation of the 0-1 loss function. With a minor modification, the method can also find the maximum likelihood (ML) clustering.

When seeking the MAP (or ML) clustering, an exhaustive evaluation of every possible clustering is impossible except in trivially-small problems. The number of possible clusterings of n items grows exponentially according to $B(n)$, the Bell number for n items (Bell 1934; Rota 1964). For example, a naive exhaustive search for a sample size of $n = 200$ would require more than $B(n) > 10^{275}$ evaluations of a density (one for each clustering). A stochastic algorithm could be used to approximate the mode, but this typically requires major computational resources and does not provide any guarantee of attaining the mode. In contrast, the algorithm proposed in this paper requires only $n(n+1)/2$ evaluations and is guaranteed to find the modal clustering. When $n = 200$, for example, the proposed method finds the mode in only 20,100 density evaluations,

*Department of Statistics, Texas A&M, University, College Station, TX, http://www.stat.tamu.edu/~dahl

 DOI:10.1214/09-BA409

which takes a fraction of a second on a common desktop computer. The proposed algorithm is feasible for much larger datasets, providing a cluster estimate for tens of thousands of items in seconds or minutes.

The method assumes an univariate sufficient statistic for each item and exploits properties of product partition models. Product partition models (reviewed in Section 2.1) have the feature that both the likelihood and the prior distribution (up to a normalizing constant) are products over components. Hence, the posterior distribution of a random partition is proportional to a product over partition components. These facts, together with some regularity conditions on the component likelihood and prior, motivate the deterministic search algorithm. Section 2.2 shows that conjugate Dirichlet process mixture (DPM) models, a popular class of Bayesian nonparametric models, are related to product partition models (Quintana and Iglesias 2003). Specifically, integrating over the latent model parameters in a conjugate DPM model leads to a product partition model with a particular cohesion.

The MAP clustering is often used as a clustering estimate in Bayesian model-based clustering procedures (e.g. Broët et al. 2002; Kim et al. 2006; Li et al. 2007). From a decision theory perspective, the MAP clustering is the optimal Bayes estimate of the clustering under the 0-1 loss function, where no loss is incurred if the clustering estimate equals the true clustering and a loss of one is incurred for any other clustering estimate (Bernardo and Smith 1994). Some authors have criticized the 0-1 loss function and have proposed clustering estimators based on pairwise probabilities that items belong to the same cluster. Medvedovic and Sivaganesan (2002) and Medvedovic et al. (2004) proposed a procedure based on hierarchical clustering of the pairwise probabilities. Dahl (2006) and Lau and Green (2007) proposed clustering estimates minimizing a loss function from Binder (1978). These methods require sampling from the posterior clustering distribution (through, for example, Markov chain Monte Carlo). It it shown in a simulation study and various examples that the MAP clustering is often comparable to that of other clustering methods. The key advantage of the MAP clustering in our class of models, however, is that the optimal clustering is guaranteed to be found quickly while other methods may require time-consuming posterior simulation or be infeasible.

The mode-finding algorithm itself is presented in Section 3, along with a general strategy for verifying that a particular univariate model satisfies the conditions necessary for the proposed algorithm. Section 4 provides several univariate, conjugate DPM mixture models satisfying the conditions necessary for the mode-finding algorithm. The mode-finding algorithm for these models is implemented in the "modalclust" contributed package to R (R Development Core Team 2008) available from the author's website. A simulaton study is detailed in Section 5, which compares both the quality of the clustering estimates and the CPU time to existing methods. Section 6 gives three illustrations of the mode-finding algorithm for clustering estimation, explores robustness to the setting of the hyperparameters, and compares to other methods. Section 7 provides some concluding comments.

2 Product Partition and Dirichlet Process Mixture Models

2.1 Product Partition Models

A clustering of n objects can be represented by a *set partition* $\boldsymbol{\pi} = \{S_1, \ldots, S_q\}$ of a set $S_0 = \{1, \ldots, n\}$ having the following properties: (1) $S_i \neq \emptyset$ for $i = 1, \ldots, q$, (2) $S_i \cap S_j = \emptyset$ for $i \neq j$, and (3) $\cup_{j=1}^{q} S_j = S_0$. The sets $S_1, \ldots, S_q$ are referred to as partition *components*. When $S_0 = \{1, 2, 3\}$, for example, there are five possible partitions, namely:

$$\{\{1,2,3\}\} \qquad \{\{1\},\{2,3\}\} \qquad \{\{1,2\},\{3\}\} \qquad \{\{1,3\},\{2\}\} \qquad \{\{1\},\{2\},\{3\}\}.$$

The set partition $\{1, 2, 3\}$ indicates that all three objects belong to the same component, while $\{1\}, \{2\}, \{3\}$ is the partition placing each object into its own component. The Bell number (Bell 1934; Rota 1964) $B(n)$ is the number of possible partitions of n objects and has the recurrence relation $B(n+1) = \sum_{k=0}^{n} \binom{n}{k} B(k)$, where $B(0) = 1$.

Interest lies in probability models for sufficient statistics $\boldsymbol{y} = \{y_i \mid i \in S_0\}$ that are parameterized by a set partition $\boldsymbol{\pi}$. In the context of cluster analysis, a set partition $\boldsymbol{\pi}$ defines a *clustering* for observed sufficient statistics $\boldsymbol{y}$ and the partition components $S_1, \ldots, S_q$ are called *clusters*.

Product partition models, introduced by Hartigan (1990) and Barry and Hartigan (1992), are a class of probability models parameterized by a set partition. These models assume that items in different partition components are independent. That is, the likelihood for a partition $\boldsymbol{\pi} = \{S_1, \ldots, S_q\}$ with observed sufficient statistics $\boldsymbol{y} = (y_1, \ldots, y_n)$ is a product over components:

$$p(\boldsymbol{y}|\boldsymbol{\pi}) = \prod_{j=1}^{q} f(\boldsymbol{y}_{S_j}), \tag{1}$$

where $\boldsymbol{y}_{S_j}$ is the vector of observations corresponding to the items of component S_j. The *component likelihood* $f(\boldsymbol{y}_S)$ — that is, the likelihood contribution from a component S — is defined for any non-empty component $S \subset S_0$ and can take any form. The partition $\boldsymbol{\pi}$ is the only parameter under consideration. Any other parameters that may have been involved in the model have been integrated over their prior. One consequence of eliminating other parameters is that the issue of bias estimators of the mixture parameters is completely avoided (Bryant and Williamson 1978, 1986; Celeux and Govaert 1993). Specific examples of $f(\boldsymbol{y}_S)$ are given in Section 4.

The prior distribution for a partition $\boldsymbol{\pi}$ is also taken to be a product over the partition components (up to a normalizing constant):

$$p(\boldsymbol{\pi}) \propto \prod_{j=1}^{q} h(S_j), \tag{2}$$

where $h(S_j) \geq 0$ is defined for each non-empty $S \subset S_0$ and is called the component *cohesion*. (Throughout the text, the symbol "$\propto$" denotes proportionality as a function of the partition $\boldsymbol{\pi}$.) The mode-finding algorithm is applicable for several classes of cohesions $h(S)$. This paper discusses three instances: (1) $h(S) = 1$, which gives a uniform prior over all set partitions, (2) $h(S) = \lambda$, which treats clusters equally regardless of their size and favors models with few clusters when $\lambda < 1$, and (3) $h(S) = \eta_0\Gamma(|S|)$, which gives the prior associated with conjugate Dirichlet process mixture (DPM) models given below.

All inference concerning $\boldsymbol{\pi}$ is made from the posterior distribution $p(\boldsymbol{\pi}|\boldsymbol{y})$. By Bayes theorem, the posterior distribution is proportional to a product over partition components:

$$p(\boldsymbol{\pi}|\boldsymbol{y}) \;\propto\; p(\boldsymbol{y}|\boldsymbol{\pi})p(\boldsymbol{\pi}) \;\propto\; \Big[\prod_{j=1}^{q} f(\boldsymbol{y}_{S_j})\Big]\Big[\prod_{j=1}^{q} h(S_j)\Big] \;=\; \prod_{j=1}^{q} f(\boldsymbol{y}_{S_j})h(S_j). \tag{3}$$

2.2 Dirichlet Process Mixture Models

The Dirichlet process mixture (DPM) model is a popular nonparametric Bayesian model. The model is reviewed below and shown to lead to a product partition model as a result of integrating away the model parameters. The connection between product partition models and DPM models was first shown by Quintana and Iglesias (2003).

In its simplest form, the DPM model assumes that observed data $\boldsymbol{y} = (y_1, \ldots, y_n)$ is generated from the following hierarchical model:

$$\begin{aligned} y_i \mid \theta_i &\sim p(y_i|\theta_i) \\ \theta_i \mid F &\sim F \\ F &\sim \mathrm{DP}(\eta_0 F_0), \end{aligned} \tag{4}$$

where $p(y|\theta)$ is a known parametric family of distributions (for the random variable y) indexed by θ, and $\mathrm{DP}(\eta_0 F_0)$ is the Dirichlet process (Ferguson 1973) centered about the distribution F_0 and having mass parameter $\eta_0 > 0$. The notation is meant to imply the independence relationships (e.g., y_1 given θ_1 is independent of the other y_i's, the other θ_i's, and F).

Blackwell and MacQueen (1973) show that $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_n)$ follows a general Polya urn scheme and may be represented in the following manner:

$$\begin{aligned} \theta_1 &\sim F_0 \\ \theta_i \mid \theta_1, \ldots, \theta_{i-1} &\sim \frac{\eta_0 F_0 \;+\; \sum_{j=1}^{i-1} \psi_{\theta_j}}{\eta_0 + i - 1}\,, \end{aligned} \tag{5}$$

where ψ_μ is the point mass distribution at μ. Notice that (5) implies that $\theta_1, \ldots, \theta_n$ may share values in common, a fact that is used below in an alternative parameterization of $\boldsymbol{\theta}$. The model in (4) is simplified by integrating out the random mixing distribution F

over its prior distribution in (5). Thus, the model in (4) becomes:

$$\begin{aligned} y_i \mid \theta_i &\sim p(y_i|\theta_i) \\ \boldsymbol{\theta} &\sim p(\boldsymbol{\theta}) \text{ given in (5).} \end{aligned} \tag{6}$$

An alternative parameterization of $\boldsymbol{\theta}$ is given in terms of a partition $\boldsymbol{\pi} = \{S_1, \ldots, S_q\}$ of $S_0 = \{1, \ldots, n\}$ and a vector of component model parameters $\boldsymbol{\phi} = \{\phi_1, \ldots, \phi_q\}$, where $\phi_1, \ldots, \phi_q$ are paired with $S_1, \ldots, S_q$, respectively. Equation (5) implies that the prior distribution $p(\boldsymbol{\pi})$ can be written as:

$$p(\boldsymbol{\pi}) \;=\; \eta_0^q \prod_{j=1}^{q} \Gamma(|S_j|) \Big/ \prod_{i=1}^{n} (\eta_0 + i - 1) \;\propto\; \prod_{j=1}^{q} \eta_0 \Gamma(|S_j|), \tag{7}$$

where $|S|$ is the number of items of the component S and $\Gamma(x)$ is the gamma function evaluated at x. Notice that (7) is proportional to the product over partition components as required by (2), where $h(S) = \eta_0 \Gamma(|S|)$ in the specific case of DPM models. The specification of the prior under this alternative parameterization is completed by noting that $\phi_1, \ldots, \phi_q$ are independently drawn from F_0. Thus, $\boldsymbol{\theta}$ is equivalent to $(\boldsymbol{\pi}, \boldsymbol{\phi})$ and the model in (4) and (6) may be expressed as:

$$\begin{aligned} y_i \mid \boldsymbol{\pi}, \boldsymbol{\phi} &\sim p(y_i|\; \phi_1 \mathbf{I}\{i \in S_1\} + \ldots + \phi_q \mathbf{I}\{i \in S_q\}\;) \\ \boldsymbol{\pi} &\sim p(\boldsymbol{\pi}) \text{ given in (7)} \\ \phi_j &\sim F_0, \end{aligned} \tag{8}$$

where $\mathbf{I}\{A\}$ is the indicator function for event A and $\phi_1, \ldots, \phi_q$ are independent and identically distributed F_0.

2.3 Conjugate DPM Models

If $p(y|\phi)$ and F_0 in (4) and (8) are chosen such that F_0 is conjugate to $p(y|\phi)$ in ϕ, the component model parameters $\boldsymbol{\phi}$ may be integrated away analytically. In a normal-normal DPM model, this technique was first used by MacEachern (1994) and has been shown to greatly improve the efficiency of Gibbs sampling (MacEachern 1994) and sequential importance sampling (MacEachern et al. 1999). Neal (1992) used this technique for models of categorical data.

Upon integrating out $\boldsymbol{\phi}$, the likelihood $p(\boldsymbol{y}|\boldsymbol{\pi})$ is given as a product over components in $\boldsymbol{\pi}$:

$$p(\boldsymbol{y}|\boldsymbol{\pi}) = \prod_{j=1}^{q} f(\boldsymbol{y}_{S_j}) \tag{9}$$

where:

$$f(\boldsymbol{y}_{S_j}) = \prod_{i=1}^{|S_j|} \int p(y_{S_j^i}|\phi_j) p(\phi_j | y_{S_j^1}, \ldots, y_{S_j^{i-1}}) d\phi_j \;, \tag{10}$$

where S_j^i is the i^{th} integer of S_j and $p(\phi_j|y_{S_j^1}, \ldots, y_{S_j^{i-1}})$ is the posterior distribution of a cluster model parameter ϕ_j based on data preceding $y_{S_j^i}$ and the prior F_0. Notice that (9) conforms to the likelihood definition for the product partition model in (1).

3 Mode-Finding Procedure

This section details the proposed mode-finding algorithm. First, a class of product partition models is defined. Then, the mode-finding algorithm for this class is explained. Finally, the validity of the algorithm is established and its efficiency is discussed.

3.1 Class of Models

The mode-finding algorithm applies to product partition models whose component likelihood $f(\boldsymbol{y}_S)$ in (1) has univariate sufficient statistics $\boldsymbol{y} = (y_1, \ldots, y_n)$ and which satisfy a condition given below. This condition is essential for the algorithm to be valid when finding the maximum likelihood clustering $\boldsymbol{\pi}_{\text{ML}}$, i.e., the mode of the likelihood. If a second condition also holds, the algorithm is also valid for finding the MAP clustering $\boldsymbol{\pi}_{\text{MAP}}$, i.e., the mode of the posterior distribution. Before stating these conditions, the definition of overlapping components needs to be introduced:

Definition 1 (Overlapping Components). Two partition components S_i and S_j are said to be overlapping if S_i contains an integer between the smallest and largest integers of S_j, or vice versa. For example, $S_i = \{1, 3\}$ and $S_j = \{2\}$ are overlapping components, but $S_i = \{1\}$ and $S_j = \{2, 3\}$ are not overlapping.

The conditions relevant to the mode-finding algorithm are now presented:

Condition 1. Without loss of generality, reorder the univariate sufficient statistics such that $y_1 \leq \ldots \leq y_n$. If components S_i and S_j overlap, then there exists two other components S_i^* and S_j^*, representing a reallocation of the items of S_i and S_j in which the number of items of each is preserved respectively, such that:

$$f(\boldsymbol{y}_{S_i})f(\boldsymbol{y}_{S_j}) \leq f(\boldsymbol{y}_{S_i^*})f(\boldsymbol{y}_{S_j^*}).$$

Section 4 provides three representative models that satisfy Condition 1, namely, a normal-normal model, a binomial-beta model, and a gamma-gamma model. Data analysis examples with these models are given in Section 6. For now, it is important to note an immediate implication of Condition 1 is that, for two overlapping components S_i and S_j, repeated reallocation of their items will lead to two more-likely components in which the original sizes are preserved and the components are nonoverlapping. Further, among all partitions of a given number of components with given sizes, Condition 1 implies that the mode must not contain overlapping components. Thus, the global mode of the likelihood $p(\boldsymbol{y}|\boldsymbol{\pi})$ in (1) can be found by simply considering all the partitions that do not contain overlapping components. This is a key feature of the mode-finding algorithm.

Condition 2. The cohesion $h(S)$ introduced in (2) depends, at most, only on the number of items contained in S.

Condition 2 implies that reallocating items as in Condition 1 leaves the cohesion of the components unchanged. Thus, Conditions 1 and 2 imply that there exists a global mode of the posterior distribution $p(\boldsymbol{\pi}|\boldsymbol{y})$ in (3) that does not contain overlapping clusters. The proposed mode-finding algorithm is based on this essential fact.

3.2 Algorithm

An algorithm to find $\boldsymbol{\pi}_{\text{MAP}}$, the partition that maximizes the posterior distribution $p(\boldsymbol{\pi}|\boldsymbol{y})$, is now introduced. This algorithm is valid if Conditions 1 and 2 hold. Condition 2 is satisfied for the three cohesions given in Section 2.1 and can typically be verified for others by trivial inspection. If only Condition 1 is met, the algorithm is still valid when finding $\boldsymbol{\pi}_{\text{ML}}$, the partition which maximizes the likelihood $p(\boldsymbol{y}|\boldsymbol{\pi})$. It may not be obvious how to verify Condition 1 for a particular $f(\boldsymbol{y}_S)$. Appendix 8 presents a general strategy for verifying Condition 1. Specific models for which Conditions 1 and 2 hold — i.e., for which the mode-finding algorithm applies — will be discussed in Section 4. The quality of the clustering results, when the algorithm is applied to data for which the conditions may not hold, is explored in a simulation study of Section 5.

In explaining the algorithm, the idea of an incomplete modal partition is helpful:

Definition 2 (Incomplete Modal Partition). A partition $\boldsymbol{\pi}^k_{\text{MAP}}$ of $\{1,\ldots,k\}$ is said to be an incomplete modal partition for $p(\boldsymbol{\pi}|\boldsymbol{y})$, the posterior for the partition of $\{1,\ldots,n\}$ in (3), if:

$$p(\boldsymbol{\pi} = \boldsymbol{\pi}^k \cup \{S_q\} \mid \boldsymbol{y}) \leq p(\boldsymbol{\pi} = \boldsymbol{\pi}^k_{\text{MAP}} \cup \{S_q\} \mid \boldsymbol{y})$$

for all partitions $\boldsymbol{\pi}^k$ of $\{1,\ldots,k\}$, where $S_q = \{k+1,\ldots,n\}$. An incomplete modal partition for $p(\boldsymbol{y}|\boldsymbol{\pi})$ is similarly defined and denoted $\boldsymbol{\pi}^k_{\text{ML}}$.

The incomplete modal partition $\boldsymbol{\pi}^k_{\text{MAP}}$ is the partition that maximizes (3) assuming the only items are $1,\ldots,k$ (i.e., ignoring $k+1,\ldots,n$). Note that $\boldsymbol{\pi}^n_{\text{MAP}}$ is exactly equal to $\boldsymbol{\pi}_{\text{MAP}}$, the mode of the posterior distribution $p(\boldsymbol{\pi}|\boldsymbol{y})$. Likewise, $\boldsymbol{\pi}^n_{\text{ML}}$ equals $\boldsymbol{\pi}_{\text{ML}}$, the mode of likelihood $p(\boldsymbol{y}|\boldsymbol{\pi})$.

There could be cases of multiple partitions of $\{1,\ldots,k\}$ satisfying Definition 2, although this quickly become highly unlikely as k increases. Nevertheless, these several partitions could be noted and considered in the algorithm that follows. This situation may or may not lead to multiple global modes of posterior distribution $p(\boldsymbol{\pi}|\boldsymbol{y})$, depending on whether k and $k+1$ belong to unique components in the global mode(s) of $p(\boldsymbol{\pi}|\boldsymbol{y})$. Since the posterior distribution (3) is proportional to a product over partition components, the incomplete modal partition is the optimal allocation of items $1,\ldots,k$ assuming k and $k+1$ belong to different clusters, regardless of the size of n.

The key proposition can now be stated:

Theorem 4. If Conditions 1 and 2 hold, then $\boldsymbol{\pi}^k_{\text{MAP}}$ can be found among the following

k candidates:

$$\begin{aligned}
&\{\{1,\ldots,k\}\}\\
&\boldsymbol{\pi}^{1}_{\mathrm{MAP}} \cup \{\{2,\ldots,k\}\}\\
&\vdots\\
&\boldsymbol{\pi}^{k-2}_{\mathrm{MAP}} \cup \{\{k-1,k\}\}\\
&\boldsymbol{\pi}^{k-1}_{\mathrm{MAP}} \cup \{\{k\}\}.
\end{aligned}$$

Proof. Consider the optimal allocation for the integer k in terms of Definition 2. By definition, k cannot be allocated with $\{k+1,\ldots,n\}$. Therefore, k belongs to a component of size m, where $1 \leq m \leq k$. By Conditions 1 and 2, this component containing k must be $\{k-m+1,\ldots,k\}$, otherwise k would belong to a component which overlaps with some other component. The only remaining integers that need to be allocated are $1,2,\ldots,k-m$. By Definition 2, these are optimally allocated as $\boldsymbol{\pi}^{k-m}_{\mathrm{MAP}}$. Therefore, the incomplete modal partition of $\{1,\ldots,k\}$ is $\boldsymbol{\pi}^{k-m}_{\mathrm{MAP}} \cup \{\{k-m+1,\ldots,k\}\}$, which is one of the k candidates listed in the statement of the proposition. □

Having this proposition, the mode-finding algorithm for $\boldsymbol{\pi}_{\mathrm{MAP}}$ is easily stated as:

Algorithm for Finding $\boldsymbol{\pi}_{\mathrm{MAP}}$: Note that $\boldsymbol{\pi}^{1}_{\mathrm{MAP}} = \{\{1\}\}$, by definition. For $k = 1,\ldots,n$, take the union of $\{\{k+1,\ldots,n\}\}$ with each of the k candidates for $\boldsymbol{\pi}^{k}_{\mathrm{MAP}}$ from Proposition 4 and set $\boldsymbol{\pi}^{k}_{\mathrm{MAP}}$ equal to the candidate which yields the maximum value of $p(\boldsymbol{\pi}|\boldsymbol{y})$ in (3). Upon finding $\boldsymbol{\pi}^{n}_{\mathrm{MAP}}$, note that this is indeed $\boldsymbol{\pi}_{\mathrm{MAP}}$, the maximizer of $p(\boldsymbol{\pi}|\boldsymbol{y})$.

Proposition 4 and the mode-finding algorithm relate to the mode $\boldsymbol{\pi}_{\mathrm{MAP}}$ of the posterior distribution. If Condition 2 is not met, the algorithm may still be used to find the mode of the likelihood $\boldsymbol{\pi}_{\mathrm{ML}}$ by simply setting $h(S) = 1$ (i.e., using a uniform prior on clusterings) in (3).

3.3 Efficiency

An implementation of the algorithm can be very fast since only $n(n+1)/2$ density evaluations are required, despite the much faster growth of the Bell number. This can be seen by noting that k ranges from $1,\ldots,n$ and that, for each k, there are only k candidates to consider. Section 6 provides several applications which require only seconds or minutes to find the modal clustering of thousands or tens of thousands of items.

The reference implementation — found in the "modalclust" contributed package to R (R Development Core Team 2008) on the author's website — also takes advantage of caching to avoid repeating computations. At step k, the contributions to $p(\boldsymbol{\pi}|\boldsymbol{y})$ in (3)

from items $k+1,\ldots,n$ are the same and need not be reevaluated. The k candidates for $\boldsymbol{\pi}^k_{\text{MAP}}$ are $\boldsymbol{\pi}^{l-1}_{\text{MAP}} \cup \{\{l,\ldots,k\}\}$ for $l = 1,\ldots,k$. Using the cached values from previous steps for the contributions to $p(\boldsymbol{\pi}|\boldsymbol{y})$ from $\boldsymbol{\pi}^1_{\text{MAP}},\ldots,\boldsymbol{\pi}^{k-1}_{\text{MAP}}$, the only new computations relate to the subsets $\{l,\ldots,k\}$ for $l = 1,\ldots,k$, namely: $f(\boldsymbol{y}_{\{l,\ldots,k\}})\ h(\{l,\ldots,k\})$, for $l = 1,\ldots,k$. In conjugate DPM models satisfying Conditions 1 and 2, this can be implemented such that the cost of evaluating one of the candidates is independent of k and n, yielding an $O(n^2)$ algorithm. Examining (7) and (10), it can be seen that:

$$\begin{aligned} f(\boldsymbol{y}_{\{l,\ldots,k\}})h(\{l,\ldots,k\}) = {} & f(\boldsymbol{y}_{\{l,\ldots,k-1\}})h(\{l,\ldots,k-1\}) \\ & \times \Big[\int p(y_k|\phi)p(\phi|y_1,\ldots,y_{k-1})d\phi \cdot (k-l)\Big]. \end{aligned} \tag{11}$$

Using cached values of $f(\boldsymbol{y}_{\{l,\ldots,k-1\}})h(\{l,\ldots,k-1\})$ and cached values of the associated sufficient statistics, the only new calculation is the second line of (11), whose complexity is independent of k and n.

4 Specific Models

This section details three specific models for which the mode-finding algorithm is applicable. All DPM models have the prior distribution given in (7) which satisfies Condition 2. The key to applying the algorithm to these models is verifying that $p(y|\theta)$ and F_0 yield a likelihood component in (10) satisfying Condition 1. A formal proof for one of the three models is provided in Appendix 9. Each model assumes that the sufficient statistics $\boldsymbol{y} = (y_i,\ldots,y_n)$ are univariate. Within cluster S, let $y^1_S,\ldots,y^{|S|}_S$ denote these statistics. The mode-finding algorithm for these three conjugate DPM models is implemented in the R contributed package "modalclust."

4.1 Normal-Normal Model

The normal-normal model assumes that $p(y|\phi)$ is the normal distribution with unknown mean ϕ and known variance σ^2 and F_0 is the normal distribution (for the random variable ϕ) with known mean μ and known variance τ^2. In this model:

$$p(\phi|y^1_S,\ldots,y^{i-1}_S) = \mathrm{N}\Bigg(\ \phi\ \Bigg|\ \frac{\sigma^2\mu + \tau^2\sum_{j=1}^{i-1} y^j_S}{\sigma^2 + (i-1)\tau^2}, \left(\frac{\sigma^2\tau^2}{\sigma^2 + (i-1)\tau^2}\right)^{-1}\ \Bigg),$$

where $\mathrm{N}(x|a,b)$ is the density of the normal distribution with mean a and variance b^{-1} evaluated at x. By conjugacy, $f(\boldsymbol{y}_S)$ in (10) is itself the following normal distribution:

$$f(\boldsymbol{y}_S) \ = \ \prod_{i=1}^{|S|} \mathrm{N}\Bigg(\ y^i_S\ \Bigg|\ \frac{\sigma^2\mu + \tau^2\sum_{j=1}^{i-1} y^j_S}{\sigma^2 + (i-1)\tau^2}\ , \left(\frac{\sigma^2\tau^2}{\sigma^2 + (i-1)\tau^2} + \sigma^2\right)^{-1}\ \Bigg). \tag{12}$$

Using the strategy of Appendix 8, it can be shown algebraically that $f(\boldsymbol{y}_S)$ in (12) satisfies Condition 1.

4.2 Binomial-Beta Model

The binomial-beta model assumes that $p(y|\phi)$ is the binomial distribution having unknown success probability ϕ and known number of trials N and F_0 is the beta distribution (for the random variable ϕ) with known parameters γ_0 and γ_1. In this model, $p(\phi|y_S^1,\ldots,y_S^{i-1}) = \text{Beta}(\phi|\gamma_0^*,\gamma_1^*)$, where $\gamma_0^* = \gamma_0 + \sum_{j=1}^{i-1} y_S^j$, $\gamma_1^* = \gamma_1 + (i-1)N - \sum_{j=1}^{i-1} y_S^j$, and $\text{Beta}(x|a,b)$ is the density of the beta distribution having mean $a/(a+b)$ evaluated at x. By conjugacy, $f(\boldsymbol{y}_S)$ in (10) is the product of beta-binomial distributions:

$$f(\boldsymbol{y}_S) \;=\; \prod_{i=1}^{|S|} \frac{\beta(\gamma_0^* + y_S^i, \gamma_1^* + N - y_S^i)}{\beta(\gamma_0^*,\gamma_1^*)\beta(y_S^i+1, N-y_S^i+1)(N+1)}, \tag{13}$$

where $\beta(x,z) = \Gamma(x)\Gamma(z)/\Gamma(x+z)$ is the beta function. The proof that $f(\boldsymbol{y}_S)$ in (13) satisfies Condition 1 is provided in Appendix 9.

4.3 Gamma-Gamma Model

The gamma-gamma model assumes that $p(y_i|\phi)$ is the gamma distribution having known shape a and unknown scale ϕ and F_0 is also a gamma distribution (for the random variable ϕ) with known shape a_0 and known scale ν. In this model, $p(\phi|y_S^1,\ldots,y_S^{i-1}) = \text{Gamma}(\phi|a_0+(i-1)a, \nu+\sum_{j=1}^{i-1} y_S^j)$, where $\text{Gamma}(x|a,b)$ is the density of the gamma distribution having mean ab^{-1} evaluated at x. By conjugacy, $f(\boldsymbol{y}_S)$ in (10) is:

$$f(\boldsymbol{y}_S) \;=\; \prod_{i=1}^{|S|} \frac{(y_S^i)^{a-1}(\nu+\sum_{j=1}^{i-1} y_S^j)^{a_0+(i-1)a-1}}{(\nu+\sum_{j=1}^{i} y_S^j)^{a_0+ia-1}} \frac{\Gamma(a_0+ia)}{\Gamma(a)\Gamma(a_0+(i-1)a)}. \tag{14}$$

Following Appendix 8, it can be shown that $f(\boldsymbol{y}_S)$ in (14) satisfies Condition 1.

5 Simulation Study

To investigate the quality of the clustering estimates and the associated computing time, a simulation study was conducted for the mode-finding algorithm applied to the normal-normal model of Section 4.1. The simulation suggests that, despite being based on the 0-1 loss function, the MAP clusterings are often comparable to clusterings from other algorithms.

Three data generating scenarios were considered, each being a four- or five-component mixture of normal components. The mixture weights $\boldsymbol{w}$, means $\boldsymbol{\mu}$, and standard deviations $\boldsymbol{\sigma}$ are given in Table 1. The scenarios were chosen to explore clustering performance under both favorable and unfavorable conditions for the proposed method. Scenario I is ideally suited for the normal-normal model with the DPM prior since: (1) the mixture weights yield clusterings that are typical of realizations from the Dirichlet process and much more likely under the DPM prior in (7) than under a uniform clustering prior, and (2) the assumption of equal variances of the normal-normal model is satisfied. The

variances are also equal in Scenario II, but the weights are such that the true clusters are much less likely under DPM prior than the true clusterings were in Scenario I (i.e., more than 10^{126} less likely). Finally, Scenario III examines the robustness of the modal clustering procedure since the equal variances assumption is violated *and* true clusterings are again relatively unlikely under the DPM prior. For each scenario, 1,000 random samples were obtained and the true clustering was recorded by noting the component giving rise to each data point. Each scenario was repeated 50 times.

The mode-finding algorithm was applied to the data using the cohesion from conjugate DPM models with $\eta_0 = 1.0$ (denoted "ModalClust(DPM)" in the simulation results) and the cohesion $h(S) = \lambda = 0.85^{1000}$ (denoted "ModalClust(Alt)"). The value of 0.85^{1000} for λ was found empirically to provide a compromise between fitting the data and the desire for parsimony. The hyperparameters of the normal-normal model in Section 4.1 were set as $\sigma^2 = (s/4)^2$, $\mu = \overline{x}$, and $\tau^2 = s^2$, where $\overline{x}$ and s^2 were the sample mean and variance.

The mode-finding algorithm was compared to the following clustering procedures: (1) MCLUST (Fraley and Raftery 1999), using both the equal ("E") and unequal ("V") variances specifications, as implemented in the R contributed package "mclust." BIC (Schwarz 1978) was used to choose among models with one to six mixture components. (2) Hierarchical clustering (Hartigan 1975), using Euclidean distance and agglomeration methods "complete" and "ward," as implemented in the default "stats" package of R. Estimated clusterings were obtained by cutting the tree at the true number of components. (3) k-means clustering (MacQueen 1967; Hartigan and Wong 1979), as implemented in the default "stats" package of R and setting k to the true number of components and using ten random starts.

The accuracy of the estimated clusterings to the true clusterings were accessed using the adjusted Rand index (Rand 1971; Hubert and Arabie 1985), as recommended by Milligan and Cooper (1986). Larger values for the adjusted Rand index correspond to better agreement, with 1.0 indicating perfect concordance. The simulation results are given in Table 1, including the average adjusted Rand index, the average number of (occupied) clusters, and the average CPU times with accompanying margins of error from 95% confidence intervals (if the standard error exceeded 0).

Several observations can be made from the simulation results. First, the modal clustering procedure with the DPM prior performs very well in Scenario I, as expected. The proposed procedure is among the best in Scenario II. Although some algorithms perform better, Scenario III demonstrates that the proposed procedure is somewhat robust to violations of the assumption of unequal variances. In terms of CPU time, k-means is clearly a very fast algorithm, regardless of the scenario. MCLUST and the proposed procedure are also substantially faster than the other methods. In summary, the modal clustering procedure has comparable performance in terms of the adjusted Rand index and can be substantially better than other methods.

Table 1: Simulation results from several clustering methods in three scenarios.

Scenario I: $\boldsymbol{w} = (0.60, 0.23, 0.08, 0.08, 0.01)$, $\boldsymbol{\mu} = (0.0, 2.0, 1.0, -1.0, -1.5)$
$\boldsymbol{\sigma} = (0.33, 0.33, 0.33, 0.33, 0.33)$

		Adj. Rand Index	No. Clusters	CPU Time
1.	ModalClust(DPM)	0.820 ± 0.007	4.88 ± 0.20	0.191 ± 0.016
2.	MCLUST(E)	0.733 ± 0.037	3.82 ± 0.23	0.131 ± 0.009
3.	HCLUST(complete)	0.575 ± 0.034	5.00	2.194 ± 0.008
4.	MCLUST(V)	0.543 ± 0.018	3.00	0.229 ± 0.018
5.	ModalClust(Alt)	0.490 ± 0.008	4.98 ± 0.04	0.159 ± 0.011
6.	k-means	0.488 ± 0.005	5.00	0.025 ± 0.001
7.	HCLUST(ward)	0.450 ± 0.022	5.00	2.051 ± 0.006

Scenario II: $\boldsymbol{w} = (0.25, 0.25, 0.25, 0.25)$, $\boldsymbol{\mu} = (-3, -1, 1, 3)$
$\boldsymbol{\sigma} = (0.75, 0.75, 0.75, 0.75)$

		Adj. Rand Index	No. Clusters	CPU Time
1.	k-means	0.673 ± 0.007	4.00	0.023 ± 0.001
2.	ModalClust(DPM)	0.670 ± 0.007	4.14 ± 0.10	0.183 ± 0.010
3.	MCLUST(E)	0.664 ± 0.014	3.92 ± 0.08	0.109 ± 0.006
4.	ModalClust(Alt)	0.647 ± 0.018	4.18 ± 0.11	0.156 ± 0.009
5.	HCLUST(ward)	0.601 ± 0.017	4.00	2.054 ± 0.006
6.	MCLUST(V)	0.542 ± 0.023	3.20 ± 0.13	0.161 ± 0.008
7.	HCLUST(complete)	0.510 ± 0.021	4.00	2.186 ± 0.007

Scenario III: $\boldsymbol{w} = (0.25, 0.25, 0.25, 0.25)$, $\boldsymbol{\mu} = (-3, -1, 1, 3)$
$\boldsymbol{\sigma} = (1.00, 0.25, 1.00, 0.50)$

		Adj. Rand Index	No. Clusters	CPU Time
1.	MCLUST(V)	0.793 ± 0.006	4.00	0.138 ± 0.006
2.	HCLUST(ward)	0.734 ± 0.013	4.00	2.049 ± 0.008
3.	ModalClust(Alt)	0.680 ± 0.007	4.00	0.151 ± 0.009
4.	k-means	0.680 ± 0.007	4.00	0.022 ± 0.001
5.	ModalClust(DPM)	0.629 ± 0.008	4.66 ± 0.14	0.187 ± 0.009
6.	MCLUST(E)	0.621 ± 0.009	4.20 ± 0.16	0.097 ± 0.007
7.	HCLUST(complete)	0.545 ± 0.021	4.00	2.186 ± 0.007

6 Illustrations

In this section the three models of Section 4 are used with the mode-finding algorithm to illustrate: (1) interesting applications of the algorithm, (2) computational feasibility for large datasets, (3) ability to examine multivariate aspects of the clusters, (4) robustness to the hyperparameter settings in the prior, and (5) similarities and differences with other clustering algorithms.

6.1 Normal-Normal Illustration: Differential Gene Expression

A common experimental design for microarray experiments is a replicated, two-treatment comparison. In searching for differentially expressed genes, data analysis often starts with an exploratory analysis. Broët et al. (2002) provide a clustering algorithm based on a gene-specific univariate score (i.e., the difference in the average expression in the two groups). Genes lying in small, extreme clusters provide evidence of being differentially expressed and warrant further attention. Their method is based on a Bayesian finite mixture model with an unknown number of Gaussian components and is fit using reversible-jump MCMC (Green 1995; Richardson and Green 1997). The model in Section 4.1 is similar to that of Broët et al. (2002), with the exception that the component standard deviation in the normal-normal DPM model must be known and constant for all clusters.

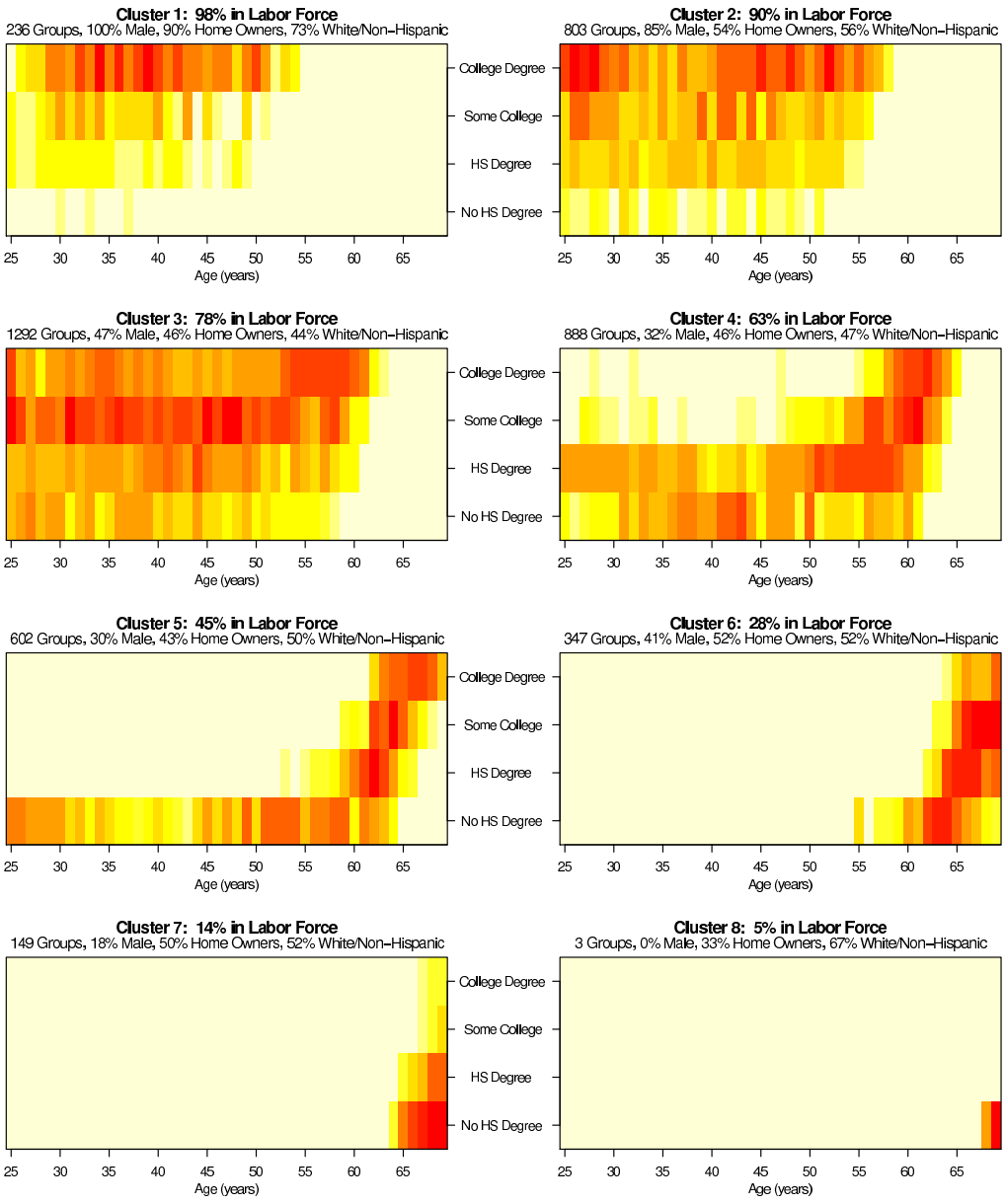
The proposed modal clustering algorithm is computationally trivial for Broët et al. (2002) dataset of 4,608 genes, taking only a couple of seconds on a common workstation. The method, however, scales well to very large datasets. Recent microarrays are able to measure the expression of an increasing number of genes. For example, the Affymetrix GeneChip HG-U95 contains over 54,000 probe sets. Applying the Bayesian finite mixture model of Broët et al. (2002) through reversible-jump MCMC for such data is computationally challenging. Even the simpler method of hierarchical clustering for such data is challenging since 11 gigabytes of RAM would be required to hold the upper-triangular part of a distance matrix of doubles for the 54,000 probe sets. In contrast, the modal clustering algorithm finds the MAP clustering in a few minutes.

6.2 Binomial-Beta Illustration: Labor Force Participation

This section uses the binomial-beta model of Section 4.2 to demonstrate that, although the algorithm is defined for univariate sufficient statistics, the multivariate aspect of the resulting clusters can be examined. Economists and policy makers are interested in studying the relationship between labor force participation and a variety of demographic variables. United States Census data was obtained from the IPUMS-USA project (http://usa.ipums.org). Random samples of 100 individuals were obtained for each of 4,320 groups defined by all possible combinations of race (non-hispanic white, other), gender (male, female), age (45 levels: 25-69 years old), educational attainment (no high school degree, high school degree, some college, college degree), home ownership status (yes, no), and census year (1980, 1990, 2000).

Using notation from Section 4.2, the number of working individuals in each group was modeled as a random realization from a binomial distribution with unknown success probability ϕ and $N = 100$ trials. The mass parameter η_0 was set to 1.0 and a uniform prior was chosen for ϕ (i.e., $\gamma_0 = 1, \gamma_1 = 1$). Note that this model assumes that the 4,320 demographic groups are exchangeable. That is, the Dirichlet process prior implies that the prior probability that any two groups are clustered is uniform across all pairs of groups. A more sophistical model might use a clustering model in which groups with similar profiles have higher *a priori* probability of belonging to the same cluster.

Figure 1: Means of demographic variables by clusters identified using the modal clustering algorithm in the labor force participation example.



Nevertheless, as seen below, the data is able to overcome the naive clustering prior and the analysis results in interpretable clusters.

The modal clustering was obtained in several seconds on a workstation. Averages for the demographic variables in each cluster are shown in Figure 1. Several interesting facts about labor force participation and the multivariate demographic variables are apparent. For example, the highest probability of labor force participation — 98% — is found in cluster 1, which contains 238 profile groups consisting of young, white, generally well-educated, males who own homes. This contrasts with, for example, cluster 5 which only has a 45% labor force participation and is made up largely of women, who are either poorly educated or close to retirement age.

In terms of the sensitivity of the results to the hyperparameters of the prior, the modal clustering was unchanged when η_0 varied from 0.1, 1.0, to 18 and used with any of the following settings: (1) $\gamma_0 = 0.1, \gamma_1 = 0.1$, (2) $\gamma_0 = 1.0, \gamma_1 = 1.0$, and (3) $\gamma_0 = 2.4, \gamma_1 = 1.2$. Further, clustering results from k-means and hierarchical clustering were obtained (as described in Section 5, except eight clusters were found to match the modal clustering). k-means was much faster than the mode-finding algorithm but inconsistent with itself — in 100 replications, the mean adjusted Rand index between two clusterings from k-means was only 0.62 ± 0.02. For the three agglomeration methods of hierarchical clustering, the clustering using "complete" agglomeration had the highest adjusted Rand index (0.76) with the modal clustering and took five times longer to compute than the modal clustering.

6.3 Gamma-Gamma Illustration: Clustering of Variances

This section demonstrates how the gamma-gamma model of Section 4.3 and the proposed modal clustering algorithm can be used to cluster variances from replicated DNA microarray experiments. Affymetrix provides test data containing 24 samples from their Human Genome U133 microarray having 22,277 probesets (simply called "genes" below). The data is background-corrected and normalized using the Robust Multichip Averaging (RMA) method of Irizarry et al. (2003) as implemented in the "affy" package of BioConductor (Gentleman et al. 2004). This multivariate data with $n = 24$ samples on $G = 22,277$ genes was reduced to sample variances $s_1^2, \ldots, s_G^2$.

The goal is to cluster sample variances $s_1^2, \ldots, s_G^2$ into groups such that genes within the same cluster are likely to have equal population variances. For each gene g, let σ_g^2 denote its population variance. Under normal theory, $(n-1)s_g^2/\sigma_g^2$ has a chi-square distribution with degrees of freedom $n-1$. This implies that s_g^2 is distributed $\text{Gamma}(s_g^2 \mid a, \theta_g)$, where $a = (n-1)/2$ is known and $\theta_g = (n-1)/(2\sigma_g^2)$ is unknown. A clustering based on equality of $\sigma_1^2, \ldots, \sigma_G^2$ is equivalent to clustering based on equality of $\theta_1, \ldots, \theta_G$. By default, let the mass parameter η_0 be 1.0. As for the hyperparameters

Table 2: Summary of clustering of variances in DNA microarray data.

Cluster	Size	Median Variance
1	49	0.00004
2	57	0.00027
3	733	0.00138
4	4644	0.00376
5	15013	0.00965
6	1747	0.02180
7	3	0.07143
8	2	0.24480
9	3	1.07472
10	26	2.70382

a_0 and ν in Section 4.3, a default choice might be:

$$\begin{aligned} a_0 &= \frac{\overline{x}^2}{s^2} + 2 \\ \nu &= \frac{2\overline{x}^3}{(n-1)s^2} + \frac{2\overline{x}}{n-1}, \end{aligned} \tag{15}$$

where $\overline{x}$ and s^2 are the sample mean and variance of the variances $s_1^2, \ldots, s_G^2$. This choice for a_0 and ν makes the prior expected value and prior variance of the unknown variance for a cluster equal to the sample mean and variance of the variances.

The mode-finding algorithm yields a clustering of the 22,277 genes in just over a minute on a standard workstation. The clustering is summarized in Table 2. The method finds a few large clusters as well as several small clusters to accommodate extreme variances. The robustness of the procedure to the specification of the hyperparameters a_0 and ν is investigated in a sensitivity analysis. Recall that $\overline{x}$ and s^2 represent the sample mean and variance of the variances $s_1^2, \ldots, s_G^2$ and are used to set a_0 and ν in (15). A sensitivity analysis investigates the concordance — in terms of the adjusted Rand index — between the clustering from the default settings and those obtained from all possible combinations of doubling and halving $\overline{x}$ and s^2 and varying the mass parameter η_0 over 0.5, 1.0, and 5.0. The results are summarized in Table 3, which indicates robustness to the mass parameter and variance, but moderate sensitivity to a_0 and ν in (15) through the mean $\overline{x}$.

Table 3: Sensitivity of results as measured by the adjusted Rand index between the modal clustering using the default settings and the modal clustering for various alternative specifications of the hyperparameters.

		Mass η_0		
Mean	Variance	0.5	1.0	5.0
$\frac{1}{2}\overline{x}$	$\frac{1}{2}s^2$	0.96	0.96	0.96
	s^2	0.96	0.96	0.96
	$2s^2$	0.96	0.96	0.96
$\overline{x}$	$\frac{1}{2}s^2$	1.00	1.00	1.00
	s^2	1.00	1.00	1.00
	$2s^2$	1.00	1.00	1.00
$2\overline{x}$	$\frac{1}{2}s^2$	0.84	0.84	0.84
	s^2	0.84	0.84	0.84
	$2s^2$	0.84	0.84	0.84

7 Conclusion

This paper defines a class of univariate product partition models and proposes a deterministic search algorithm that is guaranteed to find the MAP clustering or the ML clustering. The method is able to produce the modal clustering for thousands of items, whereas an exhaustive or stochastic search would be infeasible for all but trivially-small problems. Several univariate, conjugate DPM models satisfy the conditions necessary to apply the algorithm. The strengths of the method are its speed, simplicity, and guarantee of finding the mode. One limitation of the method is that it requires a univariate sufficient statistic for each item. Extensions to the multivariate case have been elusive.

Note our algorithm can only provide the posterior probability in (3) for the MAP partition up to a normalizing constant. When the number of items is very small (e.g., $n \leq 20$), enumerating all partitions is feasible and the posterior probability of any partition can be computed. Even so, experience shows that the posterior probability itself is not particularly interesting — it is usually very close to zero because many alternative partitions capture some of the same information seen in the MAP partition.

8 Appendix: Verifying Condition 1 for Modal Clustering

Recall that the sufficient statistics have been reordered such that $y_1 \leq \ldots \leq y_n$. Suppose that S_i and S_j are two overlapping components. Let a and c be the smallest and largest integers in S_i and let b be the smallest integer in S_j. Since S_i and S_j are overlapping components, assume (without loss of generality) that $a < b < c$ (otherwise, interchange S_i and S_j when defining a, b, and c).

Let $S_i^\flat = (S_i \setminus \{a\}) \cup \{b\}$ and $S_j^\flat = (S_j \setminus \{b\}) \cup \{a\}$. That is, define $S_i^\flat$ and $S_j^\flat$ from

S_i and S_j by simply swapping a and b. Likewise, define $S_i^\sharp$ and $S_j^\sharp$ by swapping c and b in S_i and S_j. Notice that swapping integers preserves the number of items in each component.

Condition 1 is satisfied if:

$$f(\boldsymbol{y}_{S_i})f(\boldsymbol{y}_{S_j}) \leq f(\boldsymbol{y}_{S_i^\flat})f(\boldsymbol{y}_{S_j^\flat}) \tag{16}$$

or:

$$f(\boldsymbol{y}_{S_i})f(\boldsymbol{y}_{S_j}) \leq f(\boldsymbol{y}_{S_i^\sharp})f(\boldsymbol{y}_{S_j^\sharp}). \tag{17}$$

Consider two mutually exclusive and exhaustive cases. The first case assumes that $f(\boldsymbol{y}_{S_i^\flat})f(\boldsymbol{y}_{S_j^\flat}) \leq f(\boldsymbol{y}_{S_i^\sharp})f(\boldsymbol{y}_{S_j^\sharp})$. Exploiting properties of the particular $f(\boldsymbol{y}_S)$, show that this implies (17). The second case is the complement of the first: $f(\boldsymbol{y}_{S_i^\flat})f(\boldsymbol{y}_{S_j^\flat}) > f(\boldsymbol{y}_{S_i^\sharp})f(\boldsymbol{y}_{S_j^\sharp})$. Again, using properties of the particular $f(\boldsymbol{y}_S)$, show that this second case implies (16). Thus, (16) or (17) holds and, therefore, Condition 1 is satisfied.

This strategy can be used to verify Condition 1 from each of these three models in Section 4. Implementation details are provided in Appendix 9 for one of the models.

9 Appendix: Binomial-Beta Model Satisfies Condition 1

Proof. The proof that $f(\boldsymbol{y}_S)$ of the binomial-beta model, found in (13), satisfies Condition 1 follows the notation and reasoning in Appendix 8. Two cases are considered.

Case 1. Suppose $f(\boldsymbol{y}_{S_i^\flat})f(\boldsymbol{y}_{S_j^\flat}) \leq f(\boldsymbol{y}_{S_i^\sharp})f(\boldsymbol{y}_{S_j^\sharp})$. Notice that $S_i^\flat$ and $S_j^\flat$ differ from $S_i^\sharp$ and $S_j^\sharp$ only in their allocation of a and c. (Both $S_i^\flat$ and $S_i^\sharp$ contain b.) Thus, after taking the logarithm of both sides and simplifying, the supposition of Case 1 reduces to:

$$\delta_i(y_c) + \delta_j(y_a) \leq \delta_i(y_a) + \delta_j(y_c), \tag{18}$$

where:

$$\begin{aligned}\delta_i(y) &= \ln\Gamma(\gamma_0 + \sigma_i + y) + \ln\Gamma(\gamma_1 + |S_i|\, N - \sigma_i - y)\\ \delta_j(y) &= \ln\Gamma(\gamma_0 + \sigma_j + y) + \ln\Gamma(\gamma_1 + |S_j|\, N - \sigma_j - y),\end{aligned}$$

and:

$$\sigma_i = \sum_{k=1}^{|S_i|} y_{S_i^k} + y_b - y_a - y_c \qquad\qquad \sigma_j = \sum_{k=1}^{|S_j|} y_{S_j^k} - y_b\ .$$

The goal is to show the supposition of Case 1 implies $f(\boldsymbol{y}_{S_i})f(\boldsymbol{y}_{S_j}) \leq f(\boldsymbol{y}_{S_i^\sharp})f(\boldsymbol{y}_{S_j^\sharp})$, which reduces to:

$$\delta_i(y_a + y_c - y_b) + \delta_j(y_b) \leq \delta_i(y_a) + \delta_j(y_c). \tag{19}$$

Since the gamma function is log-convex and the sum of convex functions is itself convex, $\delta_i(y)$ and $\delta_j(y)$ are convex. Also, it follows that $y_a \leq y_a + y_c - y_b \leq y_c$ since $a < b < c$ and $y_1 \leq \ldots \leq y_n$. Letting $\lambda = (y_c - y_b)/(y_c - y_a)$, the left-hand-side of (19) is:

$$\delta_i((1-\lambda)y_a + \lambda y_c) + \delta_j(\lambda y_a + (1-\lambda)y_c). \tag{20}$$

Convexity of $\delta_i(y)$ and $\delta_j(y)$ guarantees that (20) is less than or equal to:

$$\delta_i(y_a) + \delta_j(y_c) + \lambda[(\delta_i(y_c) + \delta_j(y_a)) - (\delta_i(y_a) + \delta_j(y_c))]. \tag{21}$$

By (18), the third term of (21) is non-positive and thus (21) is less than or equal to the right-hand-side of (19). Thus, (19) is established.

Case 2. Suppose $f(\boldsymbol{y}_{S_i^\flat})f(\boldsymbol{y}_{S_j^\flat}) > f(\boldsymbol{y}_{S_i^\sharp})f(\boldsymbol{y}_{S_j^\sharp})$. By similar reasoning as that of Case 1, it can be shown that this supposition implies $f(\boldsymbol{y}_{S_i})f(\boldsymbol{y}_{S_j}) \leq f(\boldsymbol{y}_{S_i^\flat})f(\boldsymbol{y}_{S_j^\flat})$.

Since one of the two cases must hold, Condition 1 is established. □

References

Barry, D. and Hartigan, J. A. (1992). "Product partition models for change point problems." *The Annals of Statistics*, 20: 260–279. 243, 245

Bell, E. T. (1934). "Exponential Numbers." *Amer. Math. Monthly*, 41(7): 411–419. 243, 245

Bernardo, J. M. and Smith, A. F. M. (1994). *Bayesian Theory*. John Wiley & Sons. 244

Binder, D. A. (1978). "Bayesian Cluster Analysis." *Biometrika*, 65: 31–38. 244

Blackwell, D. and MacQueen, J. B. (1973). "Ferguson Distributions Via Polya Urn Schemes." *The Annals of Statistics*, 1: 353–355. 246

Broët, P., Richardson, S., and Radvanyi, F. (2002). "Bayesian Hierarchical Model for Identifying Changes in Gene Expression from Microarray Experiments." *Journal of Computational Biology*, 9: 671–683. 244, 255

Bryant, P. and Williamson, J. A. (1978). "Asymptotic Behaviour of Classification Maximum Likelihood Estimates." *Biometrika*, 65: 273–282. 245

Bryant, P. G. and Williamson, J. A. (1986). "Maximum Likelihood and Classification: A Comparison of Three Approaches." In Gaul, W. and Schader, M. (eds.), *Classification as a Tool of Research*, 35–45. Elsevier/North-Holland [Elsevier Science Publishing Co., New York; North-Holland Publishing Co., Amsterdam]. 245

Celeux, G. and Govaert, G. (1993). "Comparison of the Mixture and the Classification Maximum Likelihood in Cluster Analysis." *Journal of Statistical Computation and Simulation*, 47: 127–146. 245

Dahl, D. B. (2006). "Model-Based Clustering for Expression Data via a Dirichlet Process Mixture Model." In Do, K.-A., Müller, P., and Vannucci, M. (eds.), *Bayesian Inference for Gene Expression and Proteomics*, 201–218. Cambridge University Press. 244

Ferguson, T. S. (1973). "A Bayesian Analysis of Some Nonparametric Problems." *The Annals of Statistics*, 1: 209–230. 246

Fraley, C. and Raftery, A. E. (1999). "MCLUST: Software for Model-based Cluster Analysis." *Journal of Classification*, 16(2): 297–306. 253

Gentleman, R. C., Carey, V. J., Bates, D. M., Bolstad, B., Dettling, M., Dudoit, S., Ellis, B., Gautier, L., Ge, Y., Gentry, J., Hornik, K., Hothorn, T., Huber, W., Iacus, S., Irizarry, R., Li, F. L. C., Maechler, M., Rossini, A. J., Sawitzki, G., Smith, C., Smyth, G., Tierney, L., Yang, J. Y. H., and Zhang, J. (2004). "Bioconductor: Open software development for computational biology and bioinformatics." *Genome Biology*, 5: R80.
URL http://genomebiology.com/2004/5/10/R80 257

Green, P. J. (1995). "Reversible jump Markov chain Monte Carlo computation and Bayesian model determination." *Biometrika*, 82: 711–732. 255

Hartigan, J. A. (1975). *Clustering algorithms*. John Wiley & Sons. 253

— (1990). "Partition Models." *Communications in Statistics, Part A – Theory and Methods*, 19: 2745–2756. 243, 245

Hartigan, J. A. and Wong, M. A. (1979). "[Algorithm AS 136] A K-means Clustering Algorithm (AS R39: 81V30 P355-356)." *Applied Statistics*, 28: 100–108. 253

Hubert, L. and Arabie, P. (1985). "Comparing partitions." *Journal of Classification*, 2: 193–218. 253

Irizarry, R., Hobbs, B., Collin, F., Beazer-Barclay, Y., Antonellis, K., Scherf, U., and Speed, T. (2003). "Exploration, Normalization, and Summaries of High Density Oligonucleotide Array Probe Level Data." *Biostatistics*, 4: 249–264. 257

Kim, S., Tadesse, M., and Vannucci, M. (2006). "Variable selection in clustering via Dirichlet process mixture models." *Biometrika*, 93(4): 877–893. 244

Lau, J. W. and Green, P. J. (2007). "Bayesian model based clustering procedures." *Journal of Computational and Graphical Statistics*, 16: 526–558. 244

Li, J., Ray, S., and Lindsay, B. (2007). "A Nonparametric Statistical Approach to Clustering via Mode Identication." *Journal of Machine Learning Research*, 8: 1687–1723. 244

MacEachern, S. N. (1994). "Estimating Normal Means With a Conjugate Style Dirichlet Process Prior." *Communications in Statistics, Part B – Simulation and Computation*, 23: 727–741. 247

MacEachern, S. N., Clyde, M., and Liu, J. S. (1999). "Sequential Importance Sampling for Nonparametric Bayes Models: The Next Generation." *The Canadian Journal of Statistics*, 27: 251–267. 247

MacQueen, J. B. (1967). "Some methods for classification and analysis of multivariate observations." In Le Cam, L. M. and Neyman, J. (eds.), *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, 281–297. University of California Press. 253

Medvedovic, M. and Sivaganesan, S. (2002). "Bayesian Infinite Mixture Model Based Clustering of Gene Expression Profiles." *Bioinformatrics*, 18: 1194–1206. 244

Medvedovic, M., Yeung, K., and Bumgarner, R. (2004). "Bayesian mixture model based clustering of replicated microarray data." *Bioinformatrics*, 20: 1222–1232. 244

Milligan, G. W. and Cooper, M. C. (1986). "A study of the comparability of external criteria for hierarchical cluster analysis." *Multivariate Behavioral Research*, 21: 441–458. 253

Neal, R. M. (1992). "Bayesian mixture modeling." In Smith, C. R., Erickson, G. J., and Neudorfer, P. O. (eds.), *Maximum Entropy and Bayesian Methods: Proceedings of the 11th International Workshop on Maximum Entropy and Bayesian Methods of Statistical Analysis (Seattle, 1991)*, 197–211. Kluwer Academic Publishers. 247

Quintana, F. A. and Iglesias, P. L. (2003). "Bayesian Clustering and Product Partition Models." *Journal of the Royal Statistical Society, Series B, Methodological*, 65: 557–574. 244, 246

R Development Core Team (2008). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0.
URL http://www.R-project.org 244, 250

Rand, W. M. (1971). "Objective criteria for the evaluation of clustering methods." *Journal of the American Statistical Association*, 66: 846–850. 253

Richardson, S. and Green, P. J. (1997). "On Bayesian Analysis of Mixtures With An Unknown Number of Components (Disc: P758-792) (Corr: 1998V60 P661)." *Journal of the Royal Statistical Society, Series B, Methodological*, 59: 731–758. 255

Rota, G. C. (1964). "The Number of Partitions of a Set." *Amer. Math. Monthly*, 71(5): 498–504. 243, 245

Schwarz, G. (1978). "Estimating the Dimension of a Model." *The Annals of Statistics*, 6: 461–464. 253

Acknowledgments

The author thanks Michael Newton, Marina Vannucci, and Gordon Dahl, as well as the Editor-in-Chief, editors, and referees for helpful suggestions on the presentation of the work. The mode-finding algorithm is implemented in the "modalclust" package for R available from the author's website: http://www.stat.tamu.edu/~dahl