Stat Comput (2015) 25:1023–1037
DOI 10.1007/s11222-014-9471-3

Sampling from Dirichlet process mixture models with unknown concentration parameter: mixing issues in large data implementations

David I. Hastie · Silvia Liverani · Sylvia Richardson

Received: 5 April 2013 / Accepted: 4 April 2014 / Published online: 3 May 2014

Abstract We consider the question of Markov chain Monte Carlo sampling from a general stick-breaking Dirichlet process mixture model, with concentration parameter α. This paper introduces a Gibbs sampling algorithm that combines the slice sampling approach of Walker (Communications in Statistics - Simulation and Computation 36:45–54, 2007) and the retrospective sampling approach of Papaspiliopoulos and Roberts (Biometrika 95(1):169–186, 2008). Our general algorithm is implemented as efficient open source C++ software, available as an R package, and is based on a blocking strategy similar to that suggested by Papaspiliopoulos (A note on posterior sampling from Dirichlet mixture models, 2008) and implemented by Yau et al. (Journal of the Royal Statistical Society, Series B (Statistical Methodology) 73:37–57, 2011). We discuss the difficulties of achieving good mixing in MCMC samplers of this nature in large data sets and investigate sensitivity to initialisation. We additionally consider the challenges when an additional layer of hierarchy is added such that joint inference is to be made on α. We introduce a new label-switching move and compute the marginal partition posterior to help to surmount these difficulties. Our work is illustrated using a profile regression (Molitor et al. Biostatistics 11(3):484–498, 2010) application, where we demonstrate good mixing behaviour for both synthetic and real examples.

Keywords Dirichlet process · Mixture model · Profile regression · Bayesian clustering

1 Introduction

Fitting mixture distributions to model some observed data is a common inferential strategy within statistical modelling, used in applications ranging from density estimation to regression analysis. Often, the aim is not only to fit the mixture, but additionally to use the fit to guide future predictions. Approaching the task of mixture fitting from a parametric perspective, the task to accomplish is to cluster the observed data and (perhaps simultaneously) determine the cluster parameters for each mixture component. This task is significantly complicated by the need to determine the number of mixture components that should be fitted, typically requiring complicated Markov chain Monte Carlo (MCMC) methods such as reversible jump MCMC techniques (Richardson and Green 1997) or related approaches involving parallel tempering methods (Jasra et al. 2005).

An increasingly popular alternative approach to parametric modelling is to adopt a Bayesian non-parametric approach, fitting an infinite mixture, thereby avoiding determination of the number of clusters. The Dirichlet process (Ferguson 1973) is a well studied stochastic process that is widely used in Bayesian non-parametric modelling, with particular applicability for mixture modelling. The use of the Dirichlet process in the context of mixture modelling is the basis of this paper and we shall refer to the underlying model as the Dirichlet process mixture model, or DPMM for brevity.

The idea of sampling from the DPMM is not new and has been considered by a number of authors includ-

Electronic supplementary material The online version of this article (doi:10.1007/s11222-014-9471-3) contains supplementary material, which is available to authorized users.

David I. Hastie and Silvia Liverani Joint first authors.

D. I. Hastie · S. Liverani
Imperial College London, London, UK

S. Liverani · S. Richardson (✉)
MRC Biostatistics Unit, Cambridge, UK
e-mail: sylvia.richardson@mrc-bsu.cam.ac.uk

ing Escobar and West (1995), Neal (2000), Ishwaran and James (2001), and Yau et al. (2011). While the continual evolution of samplers might implicitly suggest potential shortcomings of previous samplers, new methods are often illustrated on synthetic or low dimensional datasets which can mask issues that might arise when using the method on problems of even modest dimension. In fact, it appears that little explicit discussion has been presented detailing the inherent difficulties of using a Gibbs (or Metropolis-within-Gibbs) sampling approach to update such a complex model space, although there are some exceptions, for example Jain and Neal (2007), in the context of adding additional split-merge type moves into their sampler.

For real (rather than synthetic) data applications of the DPMM, the state space can be highly multimodal, with well separated regions of high posterior probability coexisting, often corresponding to clusterings with different number of components. We demonstrate that such highly multimodal spaces present difficulties for the existing sampling methods to escape the local modes, with poor mixing resulting in inference that is influenced by sampler initialisation. In the most serious case, this can be interpreted as non-convergence of the MCMC sampler. A primary contribution of this paper is to demonstrate these issues, highlighting that if only certain marginals are used to determine convergence they may fail to identify any issue. To address this we introduce the *Marginal Partition Posterior* as a more robust way of monitoring convergence.

A secondary (and more subtle) mixing issue relates to the mixing across the ordering of clusters in a clustering process, when a stick breaking construction is used. As we shall detail, such issues are particularly important when simultaneous inference is desired for the concentration parameter α, as defined in the following section. This mixing issue was highlighted by Papaspiliopoulos and Roberts (2008) who observed that the inclusion of label-switching moves can help to resolve the problem. We demonstrate that the moves that they propose offer only a partial solution to the problem, and we suggest an additional label-switching move that appears to enhance the performance of our own implementation of a DPMM sampler.

In the following section, we present the further details of the DPMM. Sect. 3 discusses some of the mixing issues with DPMM samplers, including Sect. 3.2 where we introduce the new label-switching move. This is followed by Sect. 4 where we present a method that we have found useful for determining sampler convergence. The implementation of our sampler is briefly summarised in Sect. 5 before Sect. 6 demonstrates some of the earlier ideas in the context of a real data example.

2 Dirichlet process mixture models

A variety of ways have been used to show the existence of the Dirichlet Process, using a number of different formulations (Ferguson 1973; Blackwell and MacQueen 1973). In this paper we focus on Dirichlet process mixture models (DPMM), based upon the following constructive definition of the Dirichlet process, due to Sethuraman (1994). If

$$\begin{aligned} P &= \sum_{c=1}^{\infty} \psi_c \delta_{\Theta_c}, \\ \Theta_c &\sim P_{\Theta_0} \quad \text{for } c \in \mathbb{Z}^+, \\ \psi_c &= V_c \prod_{l<c} (1 - V_l) \quad \text{for } c \in \mathbb{Z}^+ \backslash \{1\}, \\ \psi_1 &= V_1, \quad \text{and} \\ V_c &\sim \text{Beta}(1, \alpha) \quad \text{for } c \in \mathbb{Z}^+, \end{aligned} \tag{1}$$

where δ_x denotes the Dirac delta function concentrated at x, then $P \sim \text{DP}(\alpha, P_{\Theta_0})$. This formulation for $\boldsymbol{V}$ and $\boldsymbol{\psi}$ is known as a *stick-breaking* distribution. Importantly, the distribution P is discrete, because draws $\tilde{\Theta}_1, \tilde{\Theta}_2, \ldots$ from P can only take the values in the set $\{\Theta_c : c \in \mathbb{Z}^+\}$.

It is possible to extend the above formulation to more general stick-breaking formulations (Ishwaran and James 2001; Kalli et al. 2011; Pitman and Yor 1997).

2.1 Sampling from the DPMM

For the DPMM, the (possibly multivariate) observed data $\boldsymbol{D} = (D_1, D_2, \ldots, D_n)$ follow an infinite mixture distribution, where component c of the mixture is a parametric density of the form $f_c(\cdot) = f(\cdot|\Theta_c, \Lambda)$ parametrised by some component specific parameter Θ_c and some global parameter Λ. Defining (latent) parameters $\tilde{\Theta}_1, \tilde{\Theta}_2, \ldots, \tilde{\Theta}_n$ as draws from a probability distribution P following a Dirichlet process $DP(\alpha, P_{\Theta_0})$ and again denoting the dirac delta function by δ, this system can be written,

$$\begin{aligned} D_i|\tilde{\Theta}_i, \Lambda &\sim f(D_i|\tilde{\Theta}_i, \Lambda) \quad \text{for} \quad i = 1, 2, \ldots, n, \\ \tilde{\Theta}_i &\sim \sum_{c=1}^{\infty} \psi_c \delta_{\Theta_c} \quad \text{for} \quad i = 1, 2, \ldots, n. \end{aligned} \tag{2}$$

When making inference using mixture models (either finite or infinite) it is common practice to introduce a vector of latent allocation variables $\boldsymbol{Z}$. Such variables enable us to explicitly characterise the clustering and additionally facilitate the design of MCMC samplers. Adopting this approach and writing $\boldsymbol{\psi} = (\psi_1, \psi_2, \ldots)$ and $\boldsymbol{\Theta} = (\Theta_1, \Theta_2, \ldots)$, we re-write Eq. 2 as

$$\begin{aligned} D_i|\boldsymbol{Z}, \boldsymbol{\Theta}, \Lambda &\sim f(D_i|\Theta_{Z_i}, \Lambda) \quad \text{for} \quad i = 1, 2, \ldots, n, \\ \Theta_c &\sim P_{\Theta_0} \quad \text{for} \quad c \in \mathbb{Z}^+, \end{aligned}$$

$$\mathbb{P}(Z_i = c|\boldsymbol{\psi}) = \psi_c \quad \text{for} \quad c \in \mathbb{Z}^+, \ i = 1, 2, \ldots, n. \tag{3}$$

We refer to the model in Eq. 3 as the *full stick-breaking DPMM* or even the *FSBDPMM* for conciseness.

Historically, methods to sample from the DPMM (Escobar and West 1995; Neal 2000) have simplified the sample space of the full stick-breaking DPMM by integrating out the mixture weights $\boldsymbol{\psi}$. Collectively, such samplers have been termed *Pólya Urn* samplers. Ishwaran and James (2001) presented a number of methods for extending Pólya Urn samplers, and additionally suggested a truncation approach for sampling from the full stick-breaking DPMM with no variables integrated out.

More recently, two alternative innovative approaches to sample directly from the FSBDPMM have been proposed. The first, introduced by Walker (2007) and generalised by Kalli et al. (2011), uses a novel slice sampling approach, resulting in full conditionals that may be explored by the use of a Gibbs sampler. The second distinct MCMC sampling approach was proposed in parallel by Papaspiliopoulos and Roberts (2008). The proposed sampler again uses a Gibbs sampling approach, but is based upon an idea termed *retrospective sampling*, allowing a dynamic approach to the determination of the number of components (and their parameters) that adapts as the sampler progresses. The cost of this approach is an ingenious but complex Metropolis-within-Gibbs step, to determine cluster membership. Despite the apparent differences between the two strategies, Papaspiliopoulos (2008) noted that the two algorithms can be effectively combined to yield an algorithm that improves either of the originals. The resulting sampler was implemented and presented by Yau et al. (2011), and a similar version was used by Dunson (2009).

The current work presented in this paper uses our own sampler (described further in Sect. 5) based upon our interpretation of these ideas, implemented using our own blocking strategy. Our blocking strategy may or may not be original (we are unable to say given that the full blocking strategy adopted by Yau et al. (2011) is not explicitly detailed), but we expect our approach to be based upon a sufficiently similar strategy such that the mixing issues that we demonstrate would apply equally to other authors' implementations.

2.2 An example model

Equation 3 is of course very general, indicating that sampling from the DPMM has wide scope across a variety of applications. However, it is perhaps equally instructive to consider a specific less abstract example, that can be used to highlight the issues raised in later sections.

2.2.1 Profile regression

Recent work has used the DPMM as an alternative to parametric regression, non-parametrically linking a response vector $\boldsymbol{Y}$ with covariate data $\boldsymbol{X}$ by allocating observations to clusters. The clusters are determined by both the $\boldsymbol{X}$ and $\boldsymbol{Y}$, allowing for implicit handling of potentially high dimensional interactions which would be very difficult to capture in traditional regression. The approach also allows for the possibility of additional "fixed effects" $\boldsymbol{W}$ which have a global (i.e. non-cluster specific) effect on the response. The method is described in detail by Molitor et al. (2010), Papathomas et al. (2011), and Molitor et al. (2011), who use the term *profile regression* to refer to the approach. A similar model has independently been used by Dunson et al. (2008) and Bigelow and Dunson (2009).

Using the notation introduced earlier in this Section, the data becomes $\boldsymbol{D} = (\boldsymbol{Y}, \boldsymbol{X})$, and is modelled jointly as the product of a response model and and a covariate model resulting in the following likelihood:

$$p(D_i|Z_i, \boldsymbol{\Theta}, \Lambda, W_i) = f_Y(Y_i|\Theta_{Z_i}, \Lambda, W_i) f_X(X_i|\Theta_{Z_i}, \Lambda).$$

2.2.2 Discrete covariates with binary response

Consider the case where for each observation i, $X_i = (X_{i,1}, X_{i,2}, \ldots, X_{i,J})$ is a vector of J locally independent discrete categorical random variables, where the number of categories for covariate $j = 1, 2, \ldots, J$ is K_j. Then defining

$$\Phi_c = (\Phi_{c,1}, \Phi_{c,2} \ldots, \Phi_{c,J})$$

with $\Phi_{c,j} = (\phi_{c,j,1}, \phi_{c,j,2}, \ldots, \phi_{c,j,K_j})$, we specify the covariate model as:

$$\mathbb{P}(X_i|Z_i, \Phi_{Z_i}) = \prod_{j=1}^{J} \phi_{Z_i,j,X_{i,j}}.$$

Suppose also that Y_i is a binary response, such that

$$\text{logit}\left\{\mathbb{P}(Y_i = 1|\theta_{Z_i}, \beta, W_i)\right\} = \theta_{Z_i} + \beta^{\mathrm{T}} W_i,$$

for some vector of coefficients β.

This is simply an example of profile regression, with $\Theta_c = (\Phi_c, \theta_c)$ and $\Lambda = \beta$, such that

$$f_Y(Y_i|\Theta_{Z_i}, \Lambda, W_i) = \mathbb{P}(Y_i|\theta_{Z_i}, \beta, W_i), \quad \text{and}$$
$$f_X(X_i|\Theta_{Z_i}, \Lambda) = \mathbb{P}(X_i|Z_i, \Phi_{Z_i}).$$

We use this specific profile regression model to illustrate our results in this paper, both for the simulated dataset and the real-data example. For each cluster c we adopt the prior $\theta_c \sim \text{t}_7(0, 2.5)$ and similarly for each fixed effect l in the vector of coefficients β we adopt the prior $\beta_l \sim \text{t}_7(0, 2.5)$ while for $j = 1, \ldots, J$ we adopt the prior $\Phi_{c,j} \sim \text{Dirichlet}(a_j)$,

where a_j is a vector of 1's of length K_j. Further details about suitable prior distributions for making inference about such a model are discussed in Molitor et al. (2010) and we adopt the same priors for the examples presented below. We note however that our conclusions and the behaviour we report typically hold more broadly across the range of models that we have tested.

2.2.3 Simulated datasets

One of the key messages of our work is that DPMM samplers can perform well on simulated datasets but this does not necessarily carry through to real-data examples. We present in-depth results for a real-data example in Sect. 6, but to highlight the contrasting performance two simple simulated dataset are also used. Our first simulated data is from a profile regression model with 10 discrete covariates and a binary response variable. The dataset has 1,000 observations, partitioned at random into five groups in a balanced manner. The covariate and response distributions corresponding to each partition were selected to be well separated. The second simulated dataset is also from a profile regression model, but uses 10 discrete covariates, each with 5 categories, as well as 10 fixed effects and a Bernoulli outcome. However, in this case, the data is sampled by mixing over values of α from its Gamma prior, Gamma$(9, 0.5)$. An explicit description of the simulation methodology is provided in the Supplementary Material.

3 Mixing of MCMC algorithms for the DPMM

Sampling from a DPMM is a non-trivial exercise, as evidenced by the number of different methods that have been introduced to address a wide array of issues. For Pólya Urn samplers, with mixture weights $\boldsymbol{\psi}$ integrated out, a primary limitation is that the conditional distribution of each cluster allocation variable depends explicitly upon all other cluster allocation variables. This means that the commonly used Gibbs samplers which typically update these variables one at a time suffer from poor mixing across partition space. Using Metropolis-within-Gibbs steps and bolder split-merge moves (Jain and Neal 2004) can improve results, but in high dimensional real-data applications, designing efficient moves of this type is far from straightforward.

The challenges associated with methods which sample from the FSBDPMM (most recently Yau et al. 2011 and Kalli et al. 2011) have been perhaps less well documented. This is partially because the innovative and ingenious methods that have facilitated such sampling have required significant attention in their own right, with the consequence that the methods are often illustrated only on relatively simple datasets.

The purpose of the remainder of this Section, and the main contribution of our work, is to use our practical experience to further understanding of the behaviour of this new type of samplers, with particular emphasis on some of the challenges of sampling from the FSBDPMM for real data problems.

3.1 Initial number of clusters

A difficulty that persists even with the inclusion of the innovative techniques that allow MCMC sampling directly from the FSBDPMM is being able to effectively split clusters and thereby escape local modes. This is partially due to the intrinsic characteristics of partition spaces and the extremely high number of possible ways to split a cluster, even if it only has a relatively small number subjects in it. Although sampling directly from the FSBDPMM (rather than integrating out the mixture weights) does improve mixing when updating the allocation variables, any Gibbs moves that update allocations and parameters individually (or even in blocks) struggle to explore partition space. On the other hand, constructing more ambitious Metropolis-Hastings moves that attempt to update a larger number of parameters simultaneously is also a very difficult task due to the difficulty in designing moves to areas of the model space with similar posterior support.

Rather than subtly ignoring the problem and reporting over confident inference when analysing case studies, we suggest that, if used with caution, a FSBDPMM sampler still provides a useful inferential tool, but that its limitations must be realised and acknowledged. For example, because of the difficulty that the sampler has in increasing the number of clusters for situations involving data with weak signal, it is important to initialise the algorithm with a number of clusters which is greater than the anticipated number of clusters that the algorithm will converge to. This necessarily involves an element of trial and error to determine what that number is, where multiple runs from different initialisations must be compared (for example using the ideas presented in Sect. 4). This is demonstrated in Sect. 6.

3.2 Cluster ordering, α and label-switching

A secondary area where mixing of a full DPMM sampler requires specific attention is the mixing of the algorithm over cluster orderings. In particular, whilst the likelihood of the DPMM is invariant to the order of cluster labels, the prior specification of the stick breaking construction is not. As detailed by Papaspiliopoulos and Roberts (2008), the definition of ψ_c in terms of V_c, imposes the relation $\mathbb{E}[\psi_c] > \mathbb{E}[\psi_{c+1}]$ for all c. This weak identifiability, discussed in more detail by Porteous et al. (2006), also manifests itself through the result $P(\psi_c > \psi_{c+1}) > 0.5$ for all c, a result that we prove in Appendix 1

The importance of whether the FSBDPMM algorithm mixes sufficiently across orderings depends partially upon the object of inference. Specifically, since $P(\psi_c > \psi_{c+1})$ depends upon the prior distribution of α, if inference is to be simultaneously made about α (as is the scenario considered in this paper), it is very important that the algorithm exhibits good mixing with respect to the ordering. If this was not the case, the posterior marginal distribution for α would not be adequately sampled, and since α is directly related to the number of non-empty clusters (see Antoniak 1974 for details), this may further inhibit accurate inference being made about the number of non-empty clusters. This situation would be further exaggerated for more general stick breaking constructions (of the sort mentioned in the introduction). While it is possible to set a fixed value of α, more generally we wish to allow α to be estimated.

To ensure adequate mixing across orderings, it is important to include label-switching moves, as observed by Papaspiliopoulos and Roberts (2008). Without such moves, the one-at-a-time updates of the allocations Z_i, mean that clusters rarely switch labels, and consequentially the ordering will be largely determined by the (perhaps random) initialisation of the sampler. For all choices of α, the posterior modal ordering will be the one where the cluster with the largest number of individuals has label 1, that with the second largest has label 2 and so on. However, α affects the relative weight of other (non-modal) orderings, and a properly mixing sampler must explore these orderings according to their weights.

We adopt the label-switching moves suggested by Papaspiliopoulos and Roberts (2008), and details can be found therein. However, in our experience, while these moves may experience high acceptance rates early on in the life of the sampler, once a "good" (in terms of high posterior support) ordering is achieved, the acceptance rates drop abruptly (see Sect. 6, Fig. 7) . This means that there is little further mixing in the ordering space. Our concern is that while these label-switching moves appear to encourage a move towards the modal ordering, once that ordering is attained, the sampler rarely seems to escape too far from this ordering.

Our solution is to introduce a third label-switching move that we describe here. In brief, the idea is to simultaneously propose an update of the new cluster weights so they are something like their expected value conditional upon the new allocations. Specifically, defining $Z^\star = \max_{1\le i\le n} Z_i$ and $A = \{1, \ldots, Z^\star\}$ the move proceeds as follows: first choose a cluster c randomly from $A\backslash\{Z^\star\}$. Propose new allocations

$$Z_i' = \begin{cases} c+1 & i : Z_i = c \\ c & i : Z_i = c+1 \\ Z_i & \text{otherwise.} \end{cases} \tag{4}$$

and switch parameters associated to these clusters such that

$$\Theta_l' = \begin{cases} \Theta_{c+1} & l = c \\ \Theta_c & l = c+1 \\ \Theta_l & \text{otherwise.} \end{cases} \tag{5}$$

Additionally, propose new weights ψ_c' and ψ_{c+1}' for components c and $c+1$ such that

$$\psi_l' = \begin{cases} \psi_{c+1}\frac{\psi^+}{\Psi'}\frac{\mathbb{E}[\psi_c|\mathbf{Z}',\alpha]}{\mathbb{E}[\psi_{c+1}|\mathbf{Z},\alpha]} & l = c \\ \psi_c\frac{\psi^+}{\Psi'}\frac{\mathbb{E}[\psi_{c+1}|\mathbf{Z}',\alpha]}{\mathbb{E}[\psi_c|\mathbf{Z},\alpha]} & l = c+1 \quad \text{and} \\ \psi_l & \text{otherwise,} \end{cases} \tag{6}$$

where $\psi^+ = \psi_c + \psi_{c+1}$ and

$$\Psi' = \psi_{c+1}\frac{\mathbb{E}[\psi_c|\mathbf{Z}',\alpha]}{\mathbb{E}[\psi_{c+1}|\mathbf{Z},\alpha]} + \psi_c\frac{\mathbb{E}[\psi_{c+1}|\mathbf{Z}',\alpha]}{\mathbb{E}[\psi_c|\mathbf{Z},\alpha]},$$

by setting

$$V_l' = \begin{cases} \frac{\psi_c'}{\prod_{l<c}(1-V_l)} & l = c \\ \frac{\psi_{c+1}'}{(1-V_c')\prod_{l<c}(1-V_l)} & l = c+1 \\ V_l & \text{otherwise.} \end{cases} \tag{7}$$

All other variables are left unchanged. Assuming that there are n_c and n_{c+1} individuals in clusters c and $c+1$ respectively at the beginning of the update, the acceptance probability for this move is then given by $\min\{1, R\}$ where

$$R = \left(\frac{\psi^+}{\psi_{c+1}R_1 + \psi_c R_2}\right)^{n_c+n_{c+1}} R_1^{n_{c+1}} R_2^{n_c}, \quad \text{where} \tag{8}$$

$$R_1 = \frac{1+\alpha+n_{c+1}+\sum_{l>c+1} n_l}{\alpha+n_{c+1}+\sum_{l>c+1} n_l}, \quad \text{and} \tag{9}$$

$$R_2 = \frac{\alpha+n_c+\sum_{l>c+1} n_l}{1+\alpha+n_c+\sum_{l>c+1} n_l}. \tag{10}$$

More details can be found in Appendix 7.1.

4 Monitoring convergence

Accepting that the challenge of convergence persists, it is clearly important that the user has diagnostic methods to assess whether convergence can be reasonably expected. Due to the nature of the model space, many traditional techniques cannot be used in this context. For our hierarchical model, as described in Eqs. 1 and 3, there are no parameters that can be used to meaningfully demonstrate convergence of the algorithm. Specifically, parameters in the vector Λ tend to converge very quickly, regardless of the underlying clustering, as they are not cluster specific and therefore are not a good indication of the overall convergence. On the other

hand the cluster parameters $\boldsymbol{\Theta}_c$, cannot be tracked, as their number and interpretation changes from one iteration to the next (along with the additional complication that the labels of clusters may switch between iterations). While the concentration parameter α may appear to offer some information, using this approach can be deceiving, since a sampler that becomes stuck in a local mode in the clustering space will appear to have converged. Hence, monitoring the distribution of α across multiple runs initialised with different numbers of clusters is advisable, but in our experience finding a broad enough spectrum of initialisations is not easy to determine in advance. Therefore, relying solely on α to monitor convergence might lead to misplaced confidence.

Based upon our experience with real datasets, we suggest that to better assess convergence, it is also important to monitor the marginal partition posterior in each run, a calculation that we detail in the following section.

4.1 Marginal partition posterior

We define the marginal partition posterior as $p(\mathbf{Z}|\mathbf{D})$. This quantity represents the posterior distribution of the allocations given the data, having marginalised out all the other parameters.

In general computation of $p(\mathbf{Z}|\mathbf{D})$ is not possible in closed form, and requires certain assumptions and approximations. One such simplification is to fix the value of α in the calculation, rather than integrating over the distribution. Typically, we advise choosing one or several values of α to condition on, based on experimental runs on the dataset under study with α allowed to vary.

With the value of α fixed, whether or not $p(\mathbf{Z}|\mathbf{D})$ can be computed directly depends upon whether conjugate priors are adopted for all other parameter that must be integrated out. For the example of profile regression with logistic link introduced above this is typically not possible, as there is no natural conjugate for this response model. In such cases, integrating out such variables can be achieved using Laplace approximations. Using such an approximation appears to be sufficient for discerning differences between runs that perhaps indicate convergence problems. Details on the computations of $p(\mathbf{Z}|\mathbf{D})$ can be found in the Supplementary Material.

Figure 1 demonstrates that the strong signal in our first simulated dataset means that the sampler converges regardless of the initial number of clusters. In contrast, Sect. 6 (Fig. 2) demonstrates that for our real dataset convergence is not always achieved. For both these figures, α was fixed equal to 1.

Computing the marginal partition posterior for each run of the MCMC and comparing between runs has proven to be a very effective tool for our real examples, particularly to identify runs that were significantly different from others, perhaps due to convergence issues.

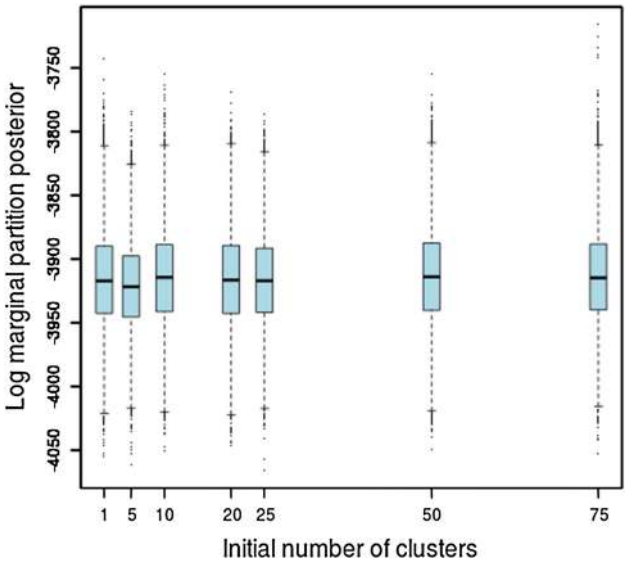

Fig. 1 Log marginal partition posterior for the first simulated dataset with different initial number of clusters and fixed $\alpha = 1$

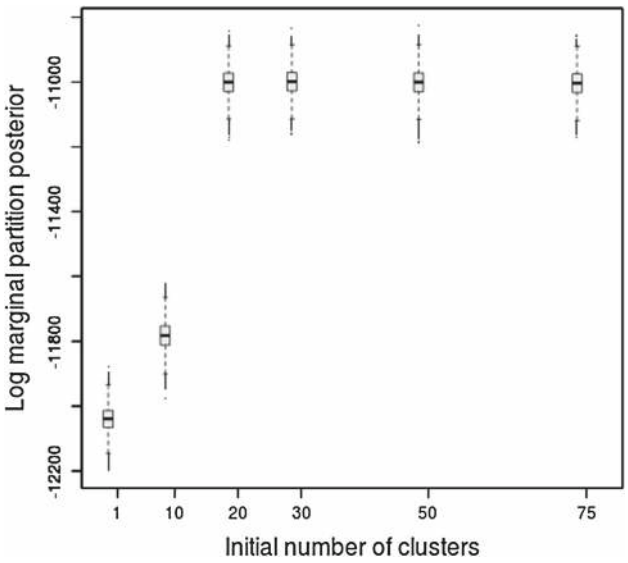

Fig. 2 Log marginal partition posterior for the real epidemiological dataset with different initial number of clusters and fixed $\alpha = 1$

Whereas comparing the marginal distribution of a parameter such as α between MCMC runs might help diagnose non-convergence if used with a wide range of initialisations, it gives no indication of which run has explored the regions of higher posterior probability. On the other hand, comparing the marginal partition posterior between two differing runs immediately indicates which run explored the higher posterior probability regions. This means that even if we are not able to make fully Bayesian inference about the parameters, we are able to draw some conclusions about those parameters which are more likely.

5 Our implementation of a DPMM sampler

To demonstrate the behaviour discussed within this paper, we have used our own implementation of a Gibbs sampler (with Metropolis-within-Gibbs steps) for the FSBDPMM. The core of the sampler is implemented as efficient `C++` code, interfaced through the `PReMiuM R` package (Liverani et al. 2013).

The sampler was originally written specifically for analysis of profile regression problems (as presented in Sect. 2.2) across a variety of applications. For such models, the package includes Bernoulli, Binomial, Poisson, Normal and categorical response models, as well as Normal and discrete covariates. It is also possible to run the sampler with no response model, allowing the consideration of more traditional mixture models. Additionally, the sampler implements a type of variable selection, allowing inference to be made in the case of data where the clustering might be determined with reference to only a subset of covariates. This type of problem is discussed in detail by Papathomas et al. (2012).

Extensive details of the algorithm can be found in (Liverani et al. 2013), including the blocking strategy that is integral for allowing sampling from the FSBDPMM. We note some brief details that are relevant to the current work below.

5.1 Post processing

5.1.1 An optimal partition

Given a sample of partitions from the posterior distribution of a Bayesian cluster model (for example from a DPMM sampler where the sample is the output of an MCMC algorithm) it is often desirable to summarise the sample as a single representative clustering estimate. The benefits of having a single estimate of the partition often sufficiently outweigh the fact that the uncertainty of the clustering is lost by such a point estimate, although it should always be communicated that this uncertainty may be considerable.

One benefit of using an optimal partition is that questions of how to account for unambiguous labelling of clusters between MCMC sweeps can be avoided. We emphasise that the term label-switching is often used in this context to refer to the complicating impact on inference of not having ways of "tracking" clusters between iterations. This is in contrast to the deliberate label-switching moves as introduced in Sect. 3.2 which use label-switching as a technique to better explore partition space and avoid undue influence of the ordering. Note that our inferential methods (e.g. determining an optimal partition or the predictive method described in the following section) are not affected by label-switching.

There are many different ways to determine a point estimate of the partition (Fritsch et al. 2009), including something as simple as the maximum a posteriori (MAP) estimate (the partition in the sample with the highest value of the marginal partition posterior). We prefer methods based on the construction (as a post-processing step) of a posterior similarity matrix, a matrix containing the posterior probabilities (estimated empirically from the MCMC run) that the observations i and j are in the same cluster. The idea is then to find a partition which maximises the sum of the pairwise similarities. We find that methods based on the posterior similarity matrix are less susceptible to Monte Carlo error than, for example, the MAP partition, especially when the optimal partition is not constrained to be in sample, but might be obtained using additional clustering methods, such as partitioning around medoids, that take advantage of the whole MCMC output. Note that once a representative partition is chosen, full uncertainty about its characteristic features can be recovered from postprocessing of the full MCMC output. See (Molitor et al. 2010) for a full discussion.

5.1.2 Making predictions

While an optimal partition can be very helpful in some cases (particularly when it is the clustering itself that is the primary object of inference) difficulties are faced in understanding or conveying the uncertainty of the partitioning. Due to the complexity and sheer size of the model space, the optimal partitions tend to differ between runs of the MCMC, and it is not an easy task to assess whether convergence has been achieved based on this approach alone.

A common target of inference is not necessarily the partition itself, but how the estimated parameters might allow us to make predictions for future observations. For example we might want to group new observations with existing observations, or, in the case of profile regression, make a prediction about the response if only the covariates of a new observation had been observed. One way to do this is to use posterior predictions, where posterior predictive distributions for quantities of interest can be derived from the whole MCMC run, taking the uncertainty over clustering into account.

Depending on the quantity of interest, the posterior predictive distribution can often be relatively robust even across runs with noticeably different optimal partitions. While this may not help us to determine if the algorithm has sufficiently explored the partition-space, if the purpose of the inference is to make predictions, this robustness can be reassuring. Moreover, by allowing predicted values to be computed based on probabilistic allocations (i.e. using a Rao-Blackwellised estimate of predictions) the sensitivity

of results to the optimal partitions of different runs is further reduced.

6 Investigation of the algorithm's properties in a large data application

In this section, we report the results of using our FSB-DPMM sampler in a profile regression application with discrete covariates and a binary response, applied to a real epidemiological dataset with 2,639 subjects.

The analysis of real data presents an important challenge: it requires care in ensuring convergence, as the signal is not as strong as in a simulation study. However, these are challenges that might be encountered more widely by users wishing to apply the methods to real data, and by presenting an example it allows us to highlight and discuss the issues that arise.

6.1 The data

Our dataset is a subset taken from an epidemiological case-control study, the analysis of which has provided the motivation of most of the work presented in this paper (see Hastie et al. 2013). In the illustrative example we have 2,639 subjects, and use 6 discrete covariates each with 5 categories, and 13 fixed effects. The response is binary and we use the model specifications detailed in Sect. 2.2 to analyse this data set. The complex epidemiological pattern in the data leads to issues with convergence of the MCMC, as we illustrate below.

Our results are based upon running multiple chains each for 100,000 iterations after a burn-in sample of 50,000 iterations. In some cases, behaviour within this burn-in period is illustrated.

6.2 Results

6.2.1 Marginal partition posterior and number of clusters

As discussed in Sect. 3 we run multiple MCMC runs, starting each with very different numbers of initial clusters. For this dataset, initialising the sampler with fewer than 20 clusters results in marginal partition posterior distributions that are significantly different between runs. This is illustrated in Fig. 2, where initialisations with small number of clusters result in much lower marginal partition posterior values than can be achieved with a higher initial number of clusters. It is apparent that there is a cut-off at around 20 clusters, where increasing the number of initial clusters further does not result in an increase in the marginal partition posterior, suggesting that with 20 clusters or more the sampler is able to visit areas of the model space with the highest posterior support.

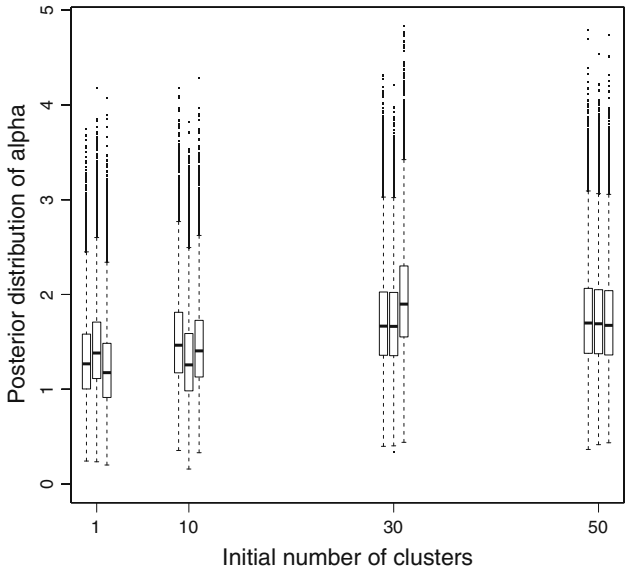

Fig. 3 Posterior distribution of α for the real epidemiological dataset for different number of initial clusters with three repetitions per initialisation: boxplots for the distribution for 50,000 sweeps after a burn-in of 50,000 samples

6.2.2 Posterior distribution of α

Figure 3 shows the boxplot of the posterior distribution of α as a function of the initial number of clusters. For each different initial number of clusters, three different runs with random initialisations of other parameters were performed. We can see that the posterior distribution of α only stabilises when the initial number of clusters is high, around 50 in our case. Thus, we would recommend carrying out such checks as part of the investigation of convergence strategy. Note that while it is advisable to start with a large number of initial clusters, starting with many more clusters than necessary can result in a larger number of iterations required for convergence.

6.2.3 Posterior distribution of the number of clusters

Figure 4 contrasts the behaviour of the sampler between the first 500 iterations of the burn in period and 500 iterations after the first 15,000, for a run with 31 initial clusters. In the initial iterations, the space is explored by modifying and merging clusters, with the number of clusters changing frequently, in a general downward trend. On the other hand, once the MCMC has converged to the model space around a mode, the algorithm attempts to split clusters regularly, but the number of changes in the number of clusters are few, and increases in the number of clusters are almost immediately reversed in the following iteration.

The need to initialise the sampler with a sufficiently high number of clusters is also supported by looking at the posterior distribution of the number of clusters. The posterior

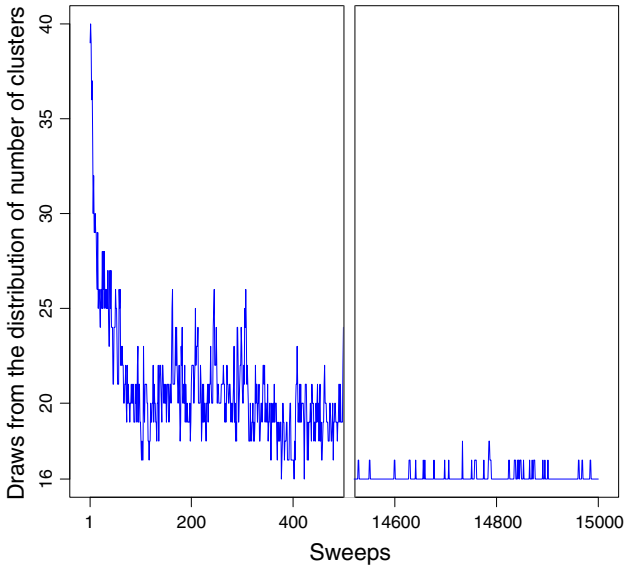

Fig. 4 The trace of the posterior of the number of clusters for the real epidemiological dataset for the first 500 iterations and after 15,000 iterations of the MCMC sampler

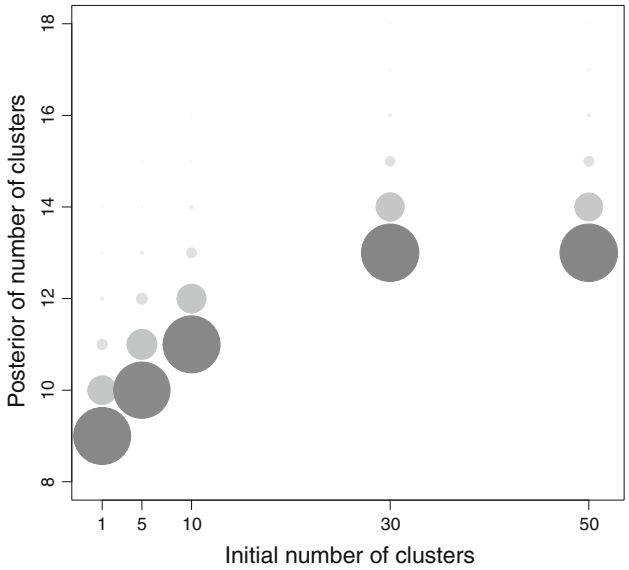

Fig. 5 The posterior distribution of the number of clusters for the real epidemiological dataset for 50,000 sweeps after a burn-in of 50,000 iterations

distributions for the number of clusters is shown in Fig. 5 for runs with different initial numbers of clusters. Five chains have been ran, initialised with 1, 5, 10, 30 and 50 clusters respectively. The size and shading of each circle in Fig. 5 represents the posterior frequency of the number of clusters for each of the chains. As can be seen from this figure, with 30 or more initial clusters the sampler has converged to a common area of posterior support, but with fewer than this the sampler might not visit this region of the model space, despite it having increased posterior support. Taken together, the plots in Figs. 2, 3 and 5 provide concurring evidence that for our real data case, starting with 50 or more clusters leads to reproducible conclusions.

6.2.4 Label-switching moves

This example also demonstrates the need for the new label-switching move discussed in Sect. 3.2 to ensure good mixing. Figure 6 demonstrates the decrease in acceptance rate that is evidenced for the label-switching moves, if only the moves that Papaspiliopoulos and Roberts (2008) propose are included. For the first of the moves that Papaspiliopoulos and Roberts (2008) propose, where the labels of two randomly selected clusters are exchanged, we observed acceptance rates below 10 % for any sample of 500 sweeps. For the second of the moves, where the labels of two neighbouring clusters are swapped, along with the corresponding V_c, V_{c+1} the acceptance rate drops considerably after initially being very high. This decrease can be explained by the observation (made by the original authors) that the second move type is always accepted if one of the clusters is empty, which can happen often in initial cluster orderings with low posterior support. Note that α stabilises after 5,000 iterations for the example shown. If only the first of the two moves is implemented, α moves extremely slowly (more than 50,000 iterations are not enough to have a stable trace; not shown) while if only the second of the two moves is implemented, for this example, 17,000 iterations are necessary for α to stabilise (not shown).

Comparing Fig. 7 to Fig. 6, we can see that the new label-switching move suffers from no drop off in acceptance at any point throughout the run. Figure 8 shows the acceptance rate for our new label-switching move, when the other two switching label moves are not included in the implementation. While the performance is worse than using all three moves, it is the most effective single label-switching move (see Sect. 3.2).

To further assess how the new label-switching move affects mixing and the ability to recover the posterior distribution of α, we used our second simulated dataset. Starting with 100 clusters, we performed 10 runs of the sampler using only moves 1 and 2 for label-switching, and 10 runs adding in our third label-switching move. In each case we ran the chain for 100,000 iterations after a burn-in sample of 100,000 iterations. Figure 9 shows the performance of the sampler in retrieving the distribution of α that was used to simulate the data with and without using our new label-switching move. It is clear that this distribution is not well recovered when using exclusively moves 1 and 2, while the addition of our third label-switching move is clearly beneficial.

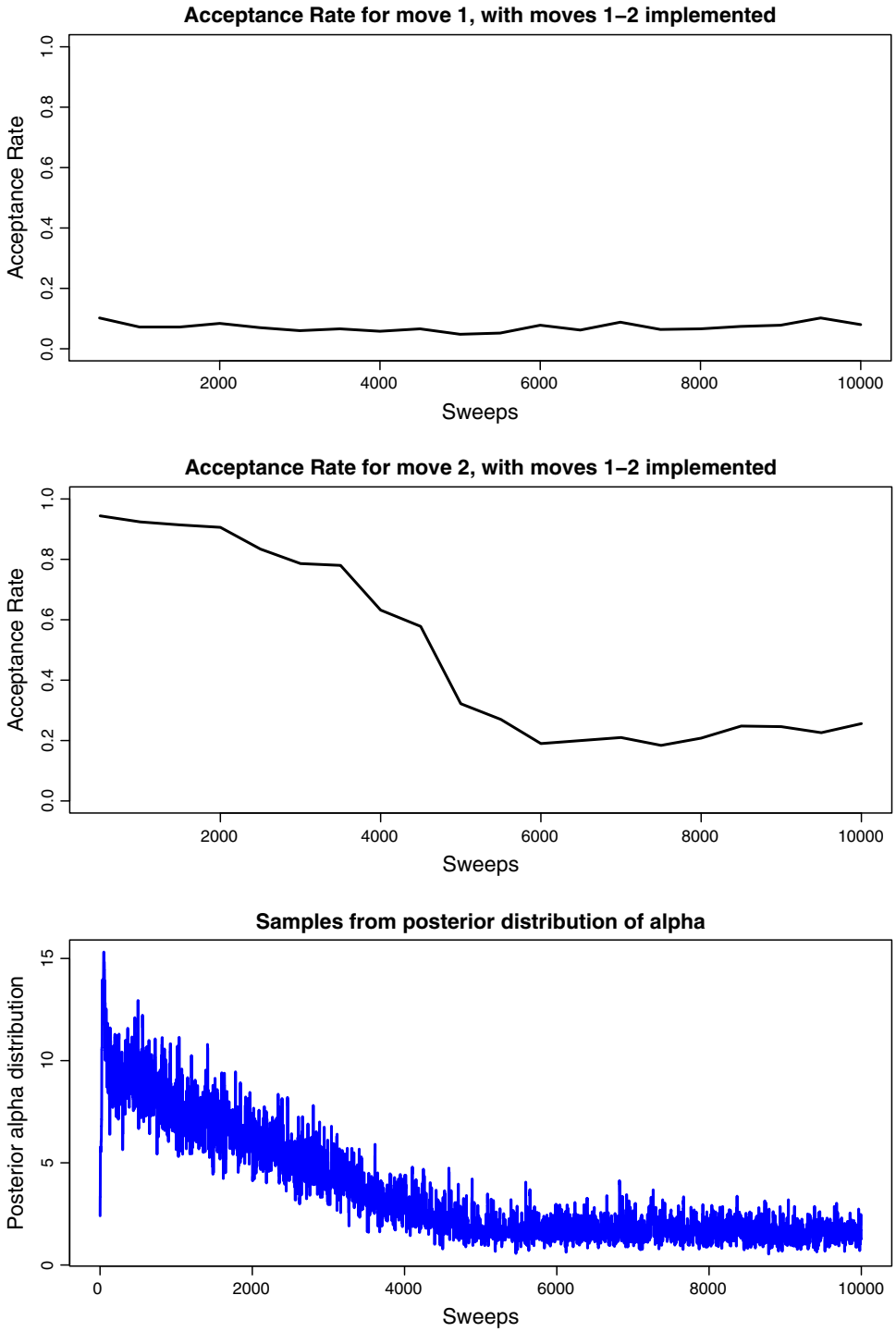

Fig. 6 Acceptance rate for intervals of 500 sweeps for the two label-switching moves proposed by Papaspiliopoulos and Roberts (2008) and comparison with samples from the posterior distribution of α (bottom) for the real epidemiological dataset

7 Conclusions

By demonstrating some of the challenges that occur when sampling from the DPMM, we hope to have raised awareness that continued research into the DPMM sampling methodology is required. Our implementation of a FSBDPMM sampler, synthesises many of the most recent and innovative techniques introduced by other authors, such as parameter blocking, slice sampling, and label-switching. However, due to the complex model space that is inherent with the FSBDPMM, many issues persist.

In previous work by other authors, considerable progress has been made evolving the samplers through innovative strategies and approaches. Nonetheless, discussion of many of the residual difficulties is avoided through demonstrating the methods only on simulated data, or for datasets with

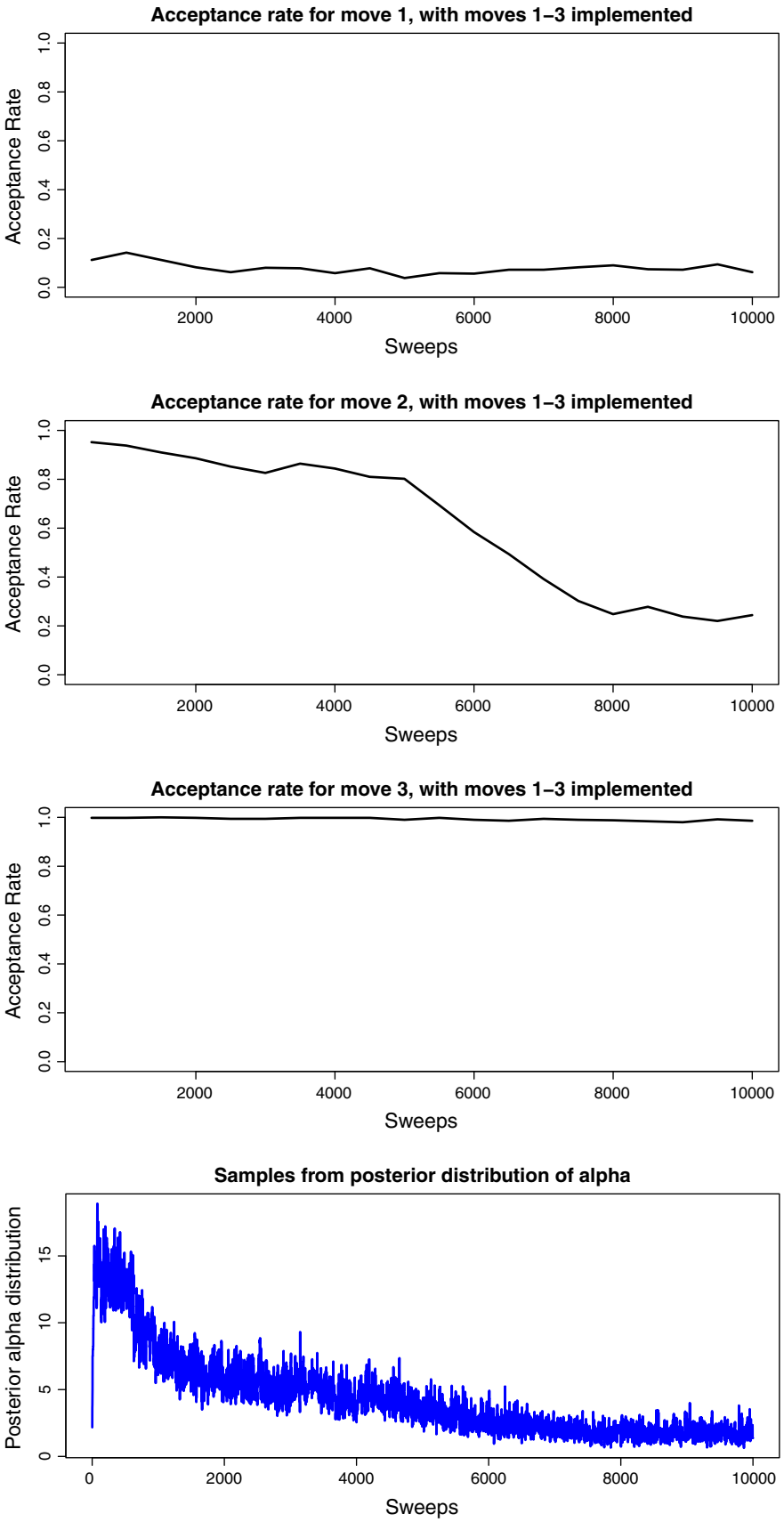

Fig. 7 Acceptance rates with the new label-switching move (Move 3) and comparison with samples from the posterior distribution of α (bottom) when all moves are implemented for the real epidemiological dataset

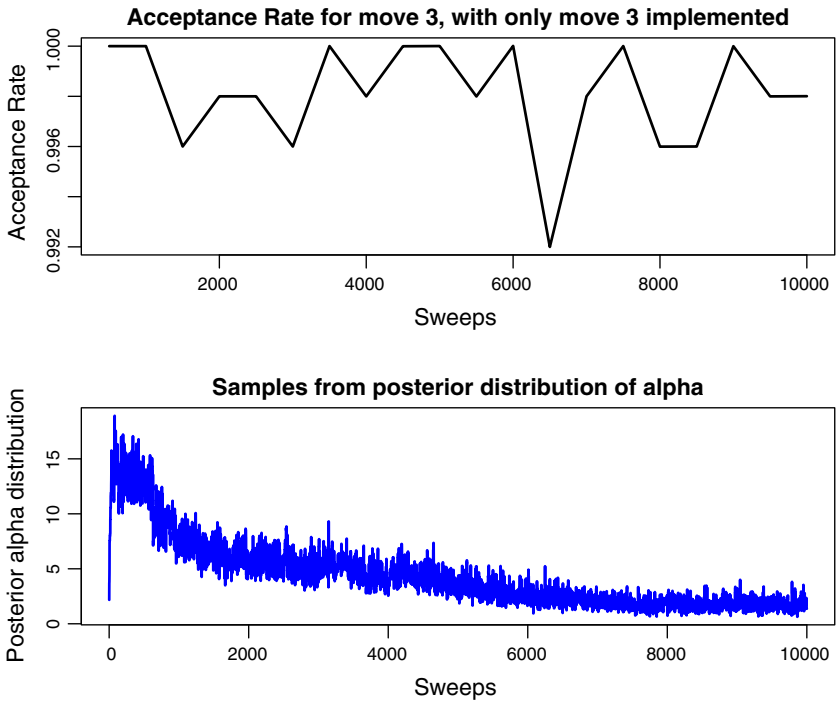

Fig. 8 Acceptance rates for the new label-switching move (Move 3) and comparison with samples from the posterior distribution of α (bottom) for the real epidemiological dataset

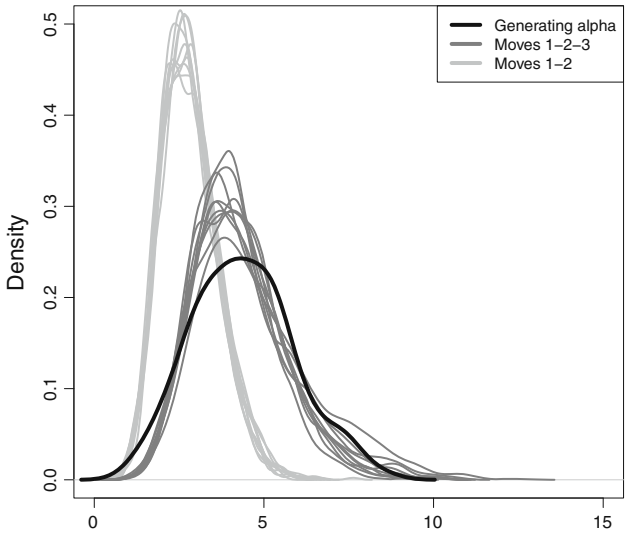

Fig. 9 Recovered posterior density of α from multiple MCMC runs with and without the new label-switching move compared with generating density of α for the second simulated dataset

strong signal. In practice however, with real datasets, the user does not have the option of simply avoiding these issues, as illustrated by our analysis of the mixing performance of an epidemiological data set with a complex epidemiological pattern.

In this paper we have attempted to highlight the difficulties that a user may face in practice. We have added a new feature in the form of an additional label-switching move to build upon this previous research and further alleviate some of the challenges that are involved when trying to sample such a complex posterior space. We have also provided practical guidelines based on our experience, on how to make useful inference in the face of these limitations.

As a consequence of discussing these challenges explicitly, we hope that our work will motivate further developments in this area to take additional steps to improve sampler efficiency. The challenge of designing MCMC moves that are able to escape local well-separated modes is considerable, but equally, so is the imagination and innovation of many researchers developing new MCMC sampling methodologies. Encouragingly research continues, and drawing on alternative techniques which might be better designed for multi-modality, such as sequential Monte Carlo (see for example Ulker et al. 2011) may yield further improvements.

In the meantime, practitioners may benefit from observing the difficulties we have presented here, allowing them to recognise and communicate potential limitations of their analyses.

Acknowledgments David I. Hastie acknowledges support from the INSERM grant (P27664). Silvia Liverani acknowledges support from the Leverhulme Trust (ECF-2011-576). Sylvia Richardson acknowledges support from MRC grant G1002319. We are grateful for helpful discussions with Sara K. Wade and Lamiae Azizi, and to Isabelle Stücker for providing the epidemiological data. We would like to the thank the editor and reviewers for their helpful comments that have allowed us to improve this paper.

Appendices

We provide the following proposition concerning the relationship between the ordering and α.

Proposition 1 *Suppose that we have a model with posterior as given in Eq. 1. Then* $\mathbb{P}(\psi_c > \psi_{c+1}|\alpha)$ *is a function of* α, *and furthermore* $\mathbb{P}(\psi_c > \psi_{c+1}) > 0.5$.

Proof If $\psi_c > \psi_{c+1}$ then $V_c > V_{c+1}(1-V_c)$, which implies $V_{c+1} < V_c/(1-V_c)$. Thus

$$\begin{aligned}
\mathbb{P}(\psi_c > \psi_{c+1}|\alpha) &= \mathbb{P}\left(V_{c+1} < V_c/(1-V_c)|\alpha\right) \\
&= \int_0^{0.5}\int_0^{V_1/(1-V_1)} \alpha^2(1-V_1)^{\alpha-1} \\
&\quad \times(1-V_2)^{\alpha-1}\mathrm{d}V_2\mathrm{d}V_1 \\
&\quad + \int_{0.5}^{1}\int_0^{1} \alpha^2(1-V_1)^{\alpha-1}(1-V_2)^{\alpha-1}\mathrm{d}V_2\mathrm{d}V_1 \\
&= \int_0^{0.5}\Big[\alpha(1-V_1)^{\alpha-1} \\
&\quad - \alpha(1-V_1)^{\alpha-1}\left(\frac{1-2V_1}{1-V_1}\right)^{\alpha}\Big]\mathrm{d}V_1 \\
&\quad + \int_{0.5}^{1}\alpha(1-V_1)^{\alpha-1}\mathrm{d}V_1 \\
&= \int_0^{1}\alpha(1-V_1)^{\alpha-1}\mathrm{d}V_1 \\
&\quad - \int_0^{0.5}\alpha\frac{(1-2V_1)^{\alpha}}{1-V_1}\mathrm{d}V_1 \\
&= 1 - \int_0^{0.5}\alpha\frac{(1-2V_1)^{\alpha}}{1-V_1}\mathrm{d}V_1.
\end{aligned}$$

Now since, $(1-2V_1)^{\alpha}/(1-V_1) < (1-2V_1)^{\alpha-1}$

$$\alpha\int_0^{0.5}\frac{(1-2V_1)^{\alpha}}{1-V_1}\mathrm{d}V_1 < \alpha\int_0^{0.5}(1-2V_1)^{\alpha-1}\mathrm{d}V_1 = 0.5.$$

So $\mathbb{P}(\psi_c > \psi_{c+1}|\alpha) > 0.5$ for all α. Finally,

$$\begin{aligned}
\mathbb{P}(\psi_c > \psi_{c+1}) &= \int \mathbb{P}(\psi_c > \psi_{c+1}|\alpha)p(\alpha)\mathrm{d}\alpha \\
&> \int 0.5p(\alpha)\mathrm{d}\alpha = 0.5.
\end{aligned}$$

□

7.1 .

Proposition 2 *Consider the label-switching move defined in Eqs. 4 to 7 in Sect. 3.2. Then:*

(i) $(\psi^+)' := \psi'_c + \psi'_{c+1} = \psi_c + \psi_{c+1} = \psi^+$;
(ii) $(1-V'_c)(1-V'_{c+1}) = (1-V_c)(1-V_{c+1})$;
(iii) The proposal mechanism is its own reverse;
(iv)

$$\frac{\mathbb{E}(\psi_c|\boldsymbol{Z}',\alpha)}{\mathbb{E}(\psi_{c+1}|\boldsymbol{Z},\alpha)} = \frac{1+\alpha+n_{c+1}+\sum_{l>c+1}n_l}{\alpha+n_{c+1}+\sum_{l>c+1}n_l} \quad \text{and}$$

$$\frac{\mathbb{E}(\psi_{c+1}|\boldsymbol{Z}',\alpha)}{\mathbb{E}(\psi_c|\boldsymbol{Z},\alpha)} = \frac{\alpha+n_c+\sum_{l>c+1}n_l}{1+\alpha+n_c+\sum_{l>c+1}n_l}; \quad \text{and}$$

(v) the acceptance probability for this move is given by $\min\{1, R\}$, *where the acceptance ratio* R *is given in Eq. 8.*

Proof (i) By definition

$$\begin{aligned}
(\psi^+)' &:= \psi'_c + \psi'_{c+1} \\
&= \frac{\psi^+}{\Psi'}\left(\psi_{c+1}\frac{\mathbb{E}[\psi_c|\boldsymbol{Z}',\alpha]}{\mathbb{E}[\psi_{c+1}|\boldsymbol{Z},\alpha]}\right. \\
&\qquad \left. + \psi_c\frac{\mathbb{E}[\psi_{c+1}|\boldsymbol{Z}',\alpha]}{\mathbb{E}[\psi_c|\boldsymbol{Z},\alpha]}\right) \\
&= \frac{\psi^+}{\Psi'}\Psi' = \psi^+;
\end{aligned}$$

(ii) From (i),

$$\psi'_c + \psi'_{c+1} = \psi_c + \psi_{c+1}$$

implies

$$\begin{aligned}
&\left[V'_c + V'_{c+1}(1-V'_c)\right]\prod_{l<c}(1-V'_l) \\
&\quad = \left[V_c + V_{c+1}(1-V_c)\right]\prod_{l<c}(1-V_l).
\end{aligned}$$

By Eq. 7, $V'_l = V_l$ for all $l < c$,

$$\begin{aligned}
&\Rightarrow \quad V'_c + V'_{c+1}(1-V'_c) = V_c + V_{c+1}(1-V_c) \\
&\Rightarrow \quad (1-V'_c)(1-V'_{c+1}) = (1-V_c)(1-V_{c+1}).
\end{aligned}$$

The importance of this result is that it provides confirmation that our proposed ψ' in Eq. 6 can be achieved with the V defined in Eq. 7. In particular, with this choice of V', the only weights that are changed are those associated with components c and $c+1$, as desired.

(iii) Suppose that the Markov chain is currently in the proposed state defined in Eqs. 4 to 7 i.e. $(\boldsymbol{V}', \boldsymbol{\Theta}', \boldsymbol{Z}', \boldsymbol{U}, \alpha, \Lambda)$. We show that applying the proposal mechanism to this state, for component c and $c+1$, the proposed new state is the original state

$$(\boldsymbol{V}'', \boldsymbol{\Theta}'', \boldsymbol{Z}'', \boldsymbol{U}, \alpha, \Lambda) = (\boldsymbol{V}, \boldsymbol{\Theta}, \boldsymbol{Z}, \boldsymbol{U}, \alpha, \Lambda.)$$

The parameters $\boldsymbol{U}$, α and Λ are unchanged by design of the proposal mechanism. Also, by design, the allocations $\boldsymbol{Z}$ and cluster parameters $\boldsymbol{\Theta}$ are simply swapped for the selected components, so trivially $\boldsymbol{Z}'' = \boldsymbol{Z}$ and $\boldsymbol{\Theta}'' = \boldsymbol{\Theta}$. Since V_l'' is unchanged for $l \notin \{c, c+1\}$, it remains only to show $V_c'' = V_c$ and $V_{c+1}'' = V_{c+1}$, or equivalently $\psi_c'' = \psi_c$ and $\psi_{c+1}'' = \psi_{c+1}$. To confirm,

$$\begin{aligned}\psi_c'' &= \psi_{c+1}' \frac{(\psi^+)'}{\Psi''} \frac{\mathbb{E}[\psi_c|\boldsymbol{Z}'']}{\mathbb{E}[\psi_{c+1}|\boldsymbol{Z}', \alpha]} \\ &= \psi_c \frac{\psi^+}{\Psi''} \frac{\psi^+}{\Psi'} \frac{\mathbb{E}[\psi_{c+1}|\boldsymbol{Z}', \alpha]}{\mathbb{E}[\psi_c|\boldsymbol{Z}, \alpha]} \frac{\mathbb{E}[\psi_c|\boldsymbol{Z}'', \alpha]}{\mathbb{E}[\psi_{c+1}|\boldsymbol{Z}', \alpha]} \\ &\quad \text{(by (i) and Equation 6)} \qquad (11) \\ &= \psi_c \frac{(\psi^+)^2}{\Psi''\Psi'} \quad \text{since } \boldsymbol{Z}'' = \boldsymbol{Z}. \qquad (12)\end{aligned}$$

However,

$$\begin{aligned}\Psi'' &= \psi_{c+1}' \frac{\mathbb{E}[\psi_c|\boldsymbol{Z}'']}{\mathbb{E}[\psi_{c+1}|\boldsymbol{Z}', \alpha]} + \psi_c' \frac{\mathbb{E}[\psi_{c+1}|\boldsymbol{Z}'', \alpha]}{\mathbb{E}[\psi_c|\boldsymbol{Z}', \alpha]} \\ &= \frac{\psi^+}{\Psi'}(\psi_c + \psi_{c+1}) \\ &\quad \text{(from Equation 6 and since } \boldsymbol{Z}'' = \boldsymbol{Z}) \\ &= \frac{(\psi^+)^2}{\Psi'}.\end{aligned}$$

Substituting this into Eq. 11 we get $\psi_c'' = \psi_c$. The result for ψ_{c+1}'' can be shown by simply following identical logic.

(iv) From Eq. 1, we have

$$\begin{aligned}\mathbb{E}[\psi_c|\boldsymbol{Z}, \alpha] &= \mathbb{E}[V_c \prod_{l<c}(1 - V_l)|\boldsymbol{Z}, \alpha] \\ &= \mathbb{E}[V_c|\boldsymbol{Z}, \alpha] \prod_{l<c} \mathbb{E}[(1 - V_l)|\boldsymbol{Z}, \alpha] \\ &= \left(\frac{1 + n_c}{1 + \alpha + n_c + \sum_{l>c} n_l}\right) \qquad (13) \\ &\quad \times \prod_{l<c} \left(\frac{\alpha + \sum_{l'>l} n_{l'}}{1 + \alpha + n_l + \sum_{l'>l} n_{l'}}\right). \qquad (14)\end{aligned}$$

Similarly,

$$\begin{aligned}\mathbb{E}[\psi_{c+1}|\boldsymbol{Z}, \alpha] &= \left(\frac{1 + n_{c+1}}{1 + \alpha + n_{c+1} + \sum_{l>c+1} n_l}\right) \\ &\quad \times \left(\frac{\alpha + \sum_{l>c} n_l}{1 + \alpha + n_c + \sum_{l>c} n_l}\right) \\ &\quad \times \prod_{l<c} \left(\frac{\alpha + \sum_{l'>l} n_{l'}}{1 + \alpha + n_l + \sum_{l'>l} n_{l'}}\right). \qquad (15)\end{aligned}$$

By definition of $\boldsymbol{Z}'$ in Eq. 4, we have

$$n_l' = \begin{cases} n_{c+1} & l = c \\ n_c & l = c+1 \\ n_l & \text{otherwise.} \end{cases} \qquad (16)$$

This means from Eqs. 13 and 15 we have

$$\begin{aligned}\frac{\mathbb{E}[\psi_c|\boldsymbol{Z}', \alpha]}{\mathbb{E}[\psi_{c+1}|\boldsymbol{Z}, \alpha]} &= \left(\frac{1 + n_c'}{1 + \alpha + n_c' + n_{c+1}' + \sum_{l>c+1} n_l}\right) \\ &\quad \times \left(\frac{1 + \alpha + n_{c+1} + \sum_{l>c+1} n_l}{1 + n_{c+1}}\right) \\ &\quad \times \left(\frac{1 + \alpha + n_c + n_{c+1} + \sum_{l>c+1} n_l}{\alpha + n_{c+1} + \sum_{l>c+1} n_l}\right) \qquad (17)\end{aligned}$$

Substituting Eq. 16 into 17 and simplifying gives the desired results. The result for $\frac{\mathbb{E}[\psi_{c+1}|\boldsymbol{Z}', \alpha]}{\mathbb{E}[\psi_c|\boldsymbol{Z}, \alpha]}$ follows in the same fashion.

(v) By (iii) and the deterministic nature of the proposal mechanism, the only random feature of the proposal is the choice of component c. The probability of this choice is the same for the move and its reverse and so cancels. Therefore the only contribution to the acceptance ratio is the ratio of posteriors. By design, the likelihood is unchanged, and by (ii) the only change in posterior is down to the change in weights of components c and $c+1$. Therefore we have,

$$\begin{aligned}R &= \frac{(\psi_c')^{n_c'}(\psi_{c+1}')^{n_{c+1}'}}{\psi_c^{n_c}\psi_{c+1}^{n_{c+1}}} \qquad (18) \\ &= \left(\frac{\psi_{c+1}'}{\psi_c}\right)^{n_c} \left(\frac{\psi_c'}{\psi_{c+1}}\right)^{n_{c+1}} \quad \text{by Equation 16.} \qquad (19)\end{aligned}$$

Substituting in Eq. 6 and the results in (iv), we obtain the desired acceptance ratio.

□

References

Antoniak, C.E.: Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems. Ann. Stat. **2**(6), 1152–1174 (1974)

Bigelow, J.L., Dunson, D.B.: Bayesian semiparametric joint models for functional predictors. J. Am. Stat. Assoc. **104**(485), 26–36 (2009)

Blackwell, D., MacQueen, J.B.: Ferguson distributions via Polya Urn Schemes. Ann. Stat. **1**(2), 353–355 (1973)

Dunson, D.B.: Nonparametric Bayes local partition models for random effects. Biometrika **96**(2), 249–262 (2009)

Dunson, D.B., Herring, A.B., Siega-Riz, A.M.: Bayesian inference on changes in response densities over predictor clusters. J. Am. Stat. Assoc. **103**(484), 1508–1517 (2008)

Escobar, M.D., West, M.: Bayesian density estimation and inference using mixtures. J. Am. Stat. Assoc. **90**(430), 577–588 (1995)

Ferguson, T.S.: A Bayesian analysis of some nonparametric problems. Ann. Stat. **1**(2), 209–230 (1973)

Fritsch, A., Ickstadt, K., et al.: Improved criteria for clustering based on the posterior similarity matrix. Bayesian anal. **4**(2), 367–391 (2009)

Hastie, D.I., Liverani, S., Azizi, L., Richardson, S., Stücker, I.: A semi-parametric approach to estimate risk functions associated with multi-dimensional exposure profiles: application to smoking and lung cancer. BMC Med. Res. Methodol. **13**, 129 (2013). doi:10.1186/1471-2288-13-129

Ishwaran, H., James, L.F.: Gibbs sampling methods for stick-breaking priors. J. Am. Stat. Assoc. **96**(453), 161–173 (2001)

Jain, S., Neal, R.M.: A split-merge Markov chain Monte Carlo procedure for the Dirichlet process mixture model. J. Comput. Graph. Stat. **13**, 158–182 (2004)

Jain, S., Neal, R.M.: Splitting and merging components of a nonconjugate dirichlet process mixture model. Bayesian Anal. **2**(3), 445–472 (2007)

Jasra, A., Holmes, C.C., Stephens, D.A.: Markov chain Monte Carlo methods and the label switching problem in Bayesian mixture modeling. Stat. Sci. **20**(1), 50–67 (2005)

Kalli, M., Griffin, J.E., Walker, S.G.: Slice sampling mixture models. Stat. Comput. **21**(1), 93–105 (2011)

Liverani, S., Hastie, D.I., Richardson, S.: PReMiuM: An R Package for Profile Regression Mixture Models using Dirichlet Processes, preprint available at arXiv:1303.2836 (2013)

Molitor, J., Papathomas, M., Jerrett, M., Richardson, S.: Bayesian profile regression with an application to the National Survey of Children's Health. Biostatistics **11**(3), 484–498 (2010)

Molitor, J., Su, J.G., Molitor, N.T., Rubio, V.G., Richardson, S., Hastie, D., Morello-Frosch, R., Jerrett, M.: Identifying vulnerable populations through an examination of the association between multipollutant profiles and poverty. Environ. Sci. Technol. **45**(18), 7754–7760 (2011)

Neal, R.M.: Markov chain sampling methods for Dirichlet process mixture models. J. Comput. Graph. Stat. **9**(2), 249 (2000)

Papaspiliopoulos, O.: A note on posterior sampling from Dirichlet mixture models. Technical Report 8, CRISM Paper (2008)

Papaspiliopoulos, O., Roberts, G.O.: Retrospective Markov chain Monte Carlo methods for Dirichlet process hierarchical models. Biometrika **95**(1), 169–186 (2008)

Papathomas, M., Molitor, J., Richardson, S., Riboli, E., Vineis, P.: Examining the joint effect of multiple risk factors using exposure risk profiles: lung cancer in non-smokers. Environ. Health Perspect. **119**, 84–91 (2011)

Papathomas, M., Molitor, J., Hoggart, C., Hastie, D.I., Richardson, S.: Exploring data from genetic association studies using Bayesian variable selection and the Dirichlet process : application to searching for gene × gene patterns. Genet. Epidemiol. **6**(36), 663–674 (2012)

Pitman, J., Yor, M.: The two-parameter Poisson-Dirichlet distribution derived from a stable subordinator. Ann. Probab. **25**(2), 855–900 (1997)

Porteous, I., Ihler, A., Smyth, P., Welling, M.: Gibbs sampling for (coupled) infinite mixture models in the stick breaking representation. In: Proceedings of the 22nd Annual Conference on Uncertainty in Artificial Intelligence (UAI-06), AUAI Press, Arlington, VA (2006)

Richardson, S., Green, P.J.: On Bayesian analysis of mixtures with an unknown number of components. J. Royal Stat. Soc., Ser. B Methodol. **59**(4), 731–792 (1997)

Sethuraman, J.: A constructive definition of Dirichlet priors. Stat. Sin. **4**, 639–650 (1994)

Ulker, Y., Gunsel, B., Cegil, A.T.: Annealed SMC samplers for non-parametric Bayesian mixture models. IEEE Signal Process. Lett. **18**, 3–6 (2011)

Walker, S.G.: Sampling the Dirichlet mixture model with slices. Commun. Stat. - Simul. Comput. **36**, 45–54 (2007)

Yau, C., Papaspiliopoulos, O., Roberts, G.O., Holmes, C.: Bayesian non-parametric hidden Markov models with applications in genomics. J. Royal Stat. Soc., Ser. B Stat. Methodol. **73**, 37–57 (2011)