Slice Sampling Mixture Models

Maria Kalli[†], Jim E. Griffin[*] & Stephen G. Walker[*]

[†] Centre for Health Services Studies, University of Kent
[*] Institute of Mathematics, Statistics & Actuarial Science, University of Kent

Abstract

We propose a more efficient version of the slice sampler for Dirichlet process mixture models described by Walker (2007). This sampler allows the fitting of infinite mixture models with a wide–range of prior specification. To illustrate this flexiblity we develop a new nonparametric prior for mixture models by normalizing an infinite sequence of independent positive random variables and show how the slice sampler can be applied to make inference in this model. Two submodels are studied in detail. The first one assumes that the positive random variables are Gamma distributed and the second assumes that they are inverse–Gaussian distributed. Both priors have two hyperparameters and we consider their effect on the prior distribution of the number of occupied clusters in a sample. Extensive computational comparisons with alternative "conditional" simulation techniques for mixture models using the standard Dirichlet process prior and our new prior are made. The properties of the new prior are illustrated on a density estimation problem.

Keywords: Dirichlet process; Markov chain Monte Carlo; Mixture model; Normalized Weights; Slice sampler.

[*] *Corresponding author*: Jim E. Griffin, Institute of Mathematics, Statistics & Actuarial Science, University of Kent, Canterbury, U. K. Tel.: +44-1227-823627; Fax: +44-1227-827932; Email: jeg28@kent.ac.uk

1 Introduction

The well known and widely used mixture of Dirichlet process (MDP) model was first introduced by Lo (1984). The MDP model, with Gaussian kernel, is given by

$$f_P(y) = \int K(y;\phi)\,\mathrm{d}P(\phi)$$

with $K(y;\phi)$ being a normal kernel and $P \sim D(M, P_0)$. We write $P \sim D(M, P_0)$ to denote that P is a Dirichlet process (Ferguson, 1973) with parameters $M > 0$, the scale parameter, and P_0, a distribution on the real line and $\phi = (\mu, \sigma^2)$ with μ to represent the mean and σ^2 the variance of the normal component. Since the advent of Markov chain Monte Carlo methods within the mainstream statistics literature (Smith and Roberts, 1993), and the specific application to the MDP model (Escobar, 1988; Escobar, 1994; Escobar and West, 1995), the model has become one of the most popular in Bayesian nonparametrics since it is possible to integrate P from the posterior defined by this model.

Variations of the original algorithm of Escobar (1988) have been numerous; for example, MacEachern (1994); Müller and MacEachern (1998); Neal (2000). All of these algorithms rely on integrating out the random distribution function from the model, removing the infinite dimensional problem. These are usually referred to as "marginal" methods. Recent ideas have left the infinite dimensional distribution in the model and found ways of sampling a sufficient but finite number of variables at each iteration of a Markov chain with the correct stationary distribution. See Papaspiliopoulos and Roberts (2008) and Walker (2007); the latter paper using slice sampling ideas. These define so–called "conditional" methods.

There has recently been interest in defining nonparametric priors for P that move beyond the Dirichlet process (see *e.g.* Lijoi *et al* (2007)) in infinite mixture models. These alternative priors allow more control over the prior cluster structure than would be possible with the Dirichlet process. The availability of computational methods for posterior inference,that do not integrate out P, allows us to implement these priors.

The purpose of this paper is two fold: 1) to develop an efficient version of the slice sampling algorithm for MDP models proposed by Walker (2007) and to extend it to more general nonparametric priors such as general stick–breaking processes and normalised weights priors and 2) to develop a new class of nonparametric prior for infinite mixture models by normalizing an infinite sequence of positive random variables, which will be termed a Normalized Weights prior. The lay–out of the paper is

as follows. In Section 2 we describe the slice–efficient sampler for the MDP model. Section 3 describes the normalized weights prior and discusses constructing a slice sampler for infinite mixture models with this prior. Section 4 discusses an application of the normalized weights prior to modelling the hazard in survival analysis and Section 5 contains numerical illustrations and an application of the normalized weight prior to density estimation. Finally, Section 6 contains conclusions and a discussion.

2 The slice–efficient sampler for the MDP

It is well known that $P \sim D(M, P_0)$ has a stick–breaking representation (Sethuraman, 1994) given by

$$P = \sum_{j=1}^{\infty} w_j \, \delta_{\phi_j},$$

where the $\{\phi_j\}$ are independent and identically distributed from P_0 and

$$w_1 = z_1, \;\; w_j = z_j \prod_{l<j} (1 - z_l)$$

with the $\{z_j\}$ being independent and identically distributed from beta$(1, M)$. It is possible to integrate P from the posterior defined by the MDP model. However, the stick–breaking representation is essential to estimation via the non–marginal methods of Papaspiliopoulos and Roberts (2008) and Walker (2007). The idea is that we can write

$$f_{z,\phi}(y) = \sum_{j=1}^{\infty} w_j \, K(y; \phi_j)$$

and the key is to find exactly which (finite number of) variables need to be sampled to produce a valid Markov chain with correct stationary distribution.

The details of the slice sampler algorithm are given in Walker (2007), but we briefly describe the basis for the algorithm here and note an improvement, also noticed by Papaspiliopoulos (2008). The joint density

$$f_{z,\phi}(y, u) = \sum_{j=1}^{\infty} \mathbf{1}(u < w_j) \, K(y; \phi_j)$$

is the starting point. Given the latent variable u, the number of mixtures is finite, the indices being $A_u = \{k : w_k > u\}$. One has

$$f_{z,\phi}(y|u) = N_u^{-1} \sum_{j \in A_u} K(y; \phi_j),$$

and the size of A_u is $N_u = \sum_{j=1}^{\infty} \mathbf{1}(w_j > u)$.

One can then introduce a further latent variable, d, which indicates which of these finite number of mixtures provides the observation to give the joint density

$$f_{z,\phi}(y, u, d) = \mathbf{1}(u < w_d)\, K(y; \phi_d).$$

Hence, a complete likelihood function for (z, ϕ) is available as a simple product of terms and crucially d is finite. Without u, d can take an infinite number of values which would make the implementation of a Markov chain Monte Carlo algorithm problematic.

We briefly describe the simulation algorithm, but only provide the sampling procedure without derivation since this has appeared elsewhere (Walker, 2007). However, as mentioned earlier, we do sample one of the full conditionals in a different and more efficient manner. We sample $\pi(z, u|\cdots)$ as a block and this involves sampling $\pi(z|\cdots \text{ exclude } u)$ and then $\pi(u|z, \cdots)$, where $\pi(z|\cdots \text{ exclude } u)$ is obtained by integrating out u from $\pi(z, u|\cdots)$. The distribution $\pi(z|\cdots \text{ exclude } u)$ will be the standard full conditional for a stick–breaking process (see Ishwaran and James (2001)). Standard MCMC theory on blocking suggests that this should lead to a more efficient sampler.

Recall that we have the model

$$f(y) = \sum_{j=1}^{\infty} w_j\, K(y; \phi_j),$$

where the $\{\phi_j\}$ are independent and identically distributed from P_0, the $\{w_j\}$ have a stick–breaking process based on the Dirichlet process, described earlier in this section.

The variables that need to be sampled at each sweep of a Gibbs sampler are

$$\{(\phi_j, z_j), j = 1, 2, \ldots; (d_i, u_i), i = 1, \ldots, n\}.$$

1. $\pi(\phi_j|\cdots) \propto p_0(\phi_j) \prod_{d_i=j} K(y_i; \phi_j)$.

2. $\pi(z_j|\cdots \text{ exclude } u) \propto \text{beta}(z_j; a_j, b_j)$, where

$$a_j = 1 + \sum_{i=1}^{n} \mathbf{1}(d_i = j)$$

and

$$b_j = M + \sum_{i=1}^{n} \mathbf{1}(d_i > j).$$

3. $\pi(u_i|\cdots) \propto \mathbf{1}(0 < u_i < w_{d_i})$.

4. $\mathrm{P}(d_i = k|\cdots) \propto \mathbf{1}(k : w_k > u_i)\, K(y_i; \phi_k)$.

Obviously, we can not sample all of the (ϕ_j, z_j). But it is not required to in order to proceed with the chain. We only need to sample up to the integer N for which we have found all the appropriate w_k in order to do step 4 exactly. Since the weights sum to 1 if we find N_i such that $\sum_{k=1}^{N_i} w_k > 1 - u_i$ then it is not possible for any of the w_k, for $k > N_i$, to be greater than u_i.

There are some important points to make here. First, it is a trivial extension to consider more general stick–breaking processes for which $z_j \sim \mathrm{beta}(\alpha_j, \beta_j)$ independently. Then, in this case, we would have

$$a_j = \alpha_j + \sum_{i=1}^{n} \mathbf{1}(d_i = j)$$

and

$$b_j = \beta_j + \sum_{i=1}^{n} \mathbf{1}(d_i > j).$$

This easy extension to more general priors is not a feature of alternative, marginal sampling algorithms. Secondly, the algorithm is remarkably simple to implement; all full conditionals are standard.

Later, for the illustrations and comparison, we will consider two types of slice sampler. The "slice–efficient" which is the one described above and the "slice" which is the original algorithm appearing in Walker (2007) and is noted by the fact that the v is sampled conditional on u in this case.

The retrospective sampler (Papaspiliopoulos and Roberts 2008) is an alternative, conditional method. The following argument gives some understanding for the difference between retrospective sampling (which uses Metropolis sampling) and slice sampling. Suppose we wish to sample from $f(x) \propto l(x)\pi(x)$ using Metropolis sampling and use $\pi(x)$ as the proposal density. Let x_c be the current sample and $x^* \sim \pi(x)$ and $u \sim \mathrm{Un}(0, 1)$, so the new sample x_n is x^* if $u < l(x^*)/l(x_c)$ or else is x_c.

On the other hand, the slice sampler would work by considering $f(x, u) \propto \mathbf{1}(u < l(x))\pi(x)$ and so a move from x_c to x_n would work by sampling x_n from $\pi(x)$ restricted to $\{x : l(x)/l(x_c) > u\}$ where $u \sim \mathrm{Un}(0, 1)$. So the two sampling strategies are using the same variables but in a fundamentally different way, which allows the slice sampling version to always move.

This illustration is obviously demonstrated on a simple level, but we believe the principle applies to the difference between the retrospective sampler and the slice sampler for the mixture of Dirichlet process model.

3 Mixtures Based on Normalized Weights

3.1 Definition and Properties

The slice sampling idea can be extended to mixture models with weights obtained via normalization. The Dirichlet process has been the dominant prior in nonparametrics but the definition of alternative nonparametric priors has been a recent area of interest. For example, Lijoi *et al* (2007) define nonparametric priors through the normalization of the generalized Gamma process. We discuss an alternative form of normalization. We consider

$$f(y) = \sum_{j=1}^{\infty} w_j \, K(y; \phi_j)$$

where $w_j = \lambda_j / \Lambda$ and $\Lambda = \sum_{j=1}^{\infty} \lambda_j$. We will also use $\Lambda_m = \sum_{j=m+1}^{\infty} \lambda_j$. Here the $\{\lambda_j\}$ are positive and will be assigned independent prior distributions, say $\lambda_j \sim \pi_j(\lambda_j)$. These must be constructed so as to ensure that $\sum_{j=1}^{\infty} \lambda_j < +\infty$ a.s. We suggest defining specific priors by defining $\mathrm{E}[\lambda_j] = \xi q_j$ where $\xi > 0$ and $q_j = P(X = j)$ where X is a random variable whose distribution is discrete on the positive integers. For example, we could assume that $X = Y + 1$ where Y follows a geometric distribution. Then

$$q_j = (1 - \theta)\theta^{j-1}.$$

The parameter θ controls the rate at which $\mathrm{E}[\lambda_1], \mathrm{E}[\lambda_2], \mathrm{E}[\lambda_2], \ldots$ tends to zero. We have defined a nonparametric prior with two parameters θ and ξ. As we will see in the following examples, the choice of the distributions $\pi_1, \pi_2, \pi_3, \ldots$ controls the properties of the process. Many other families of nonparametric prior distribution can be generated by different choices of X. For example, we could assume that $X = Y + 1$ where Y follows a Poisson distribution.

Example 1: Gamma distribution.
Here we take the $\{\lambda_j\}$ to be independent gamma distributions, say $\lambda_j \sim \mathrm{Ga}(\gamma_j, 1)$. To ensure that $\Lambda < +\infty$ a.s. we take $\sum_{j=1}^{\infty} \gamma_j < +\infty$. Clearly, w_j has expectation

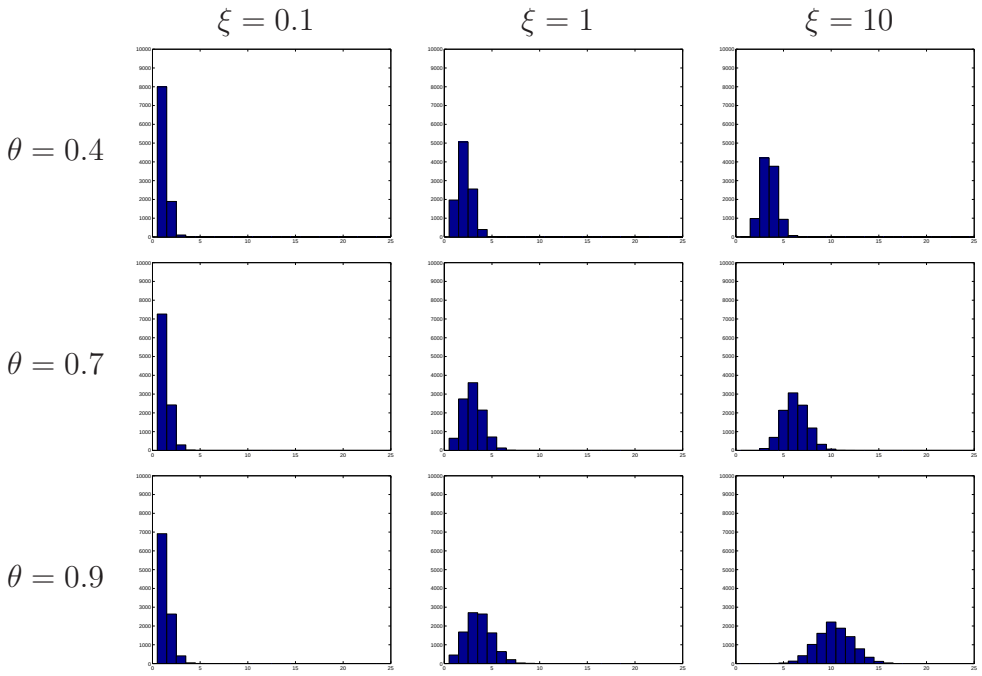

Figure 1: Prior distribution of the number of clusters from 30 observations with the infinite Dirichlet prior

q_j and variance $q_j(1 - q_j)/(\xi + 1)$ and we can interpret ξ as a mass parameter. We will refer to this model as an infinite Dirichlet prior since if we have a finite number of unnormalized weights $\lambda_1, \lambda_2, \ldots, \lambda_N$ then $w_1, w_2, \ldots, w_N$ would be Dirichlet distributed. In infinite mixture models, the prior distribution on the number of clusters from n observations is important. Figure 1 shows this distribution for $n = 30$. Larger values of θ for fixed ξ place more mass on larger numbers of clusters (as we would expect since the weights decay increasingly slowly with larger θ). The mass parameter ξ also plays an important role. Larger values of ξ lead to more dispersed distributions with a larger median value.

Stick–breaking priors were introduced to Bayesian nonparametrics by Ishwaran and James (2001). They are defined by two infinite vectors of parameters. Clearly, there is a need to develop priors within this class that have a few hyperparameters to allow easy prior specification. The Dirichlet process and Poisson-Dirichlet process are two such priors and the infinite Dirichlet prior represents another. The stick-breaking representation of the infinite Dirichlet prior takes $\alpha_j = \xi q_j$ and $\beta_j = \xi \left(1 - \sum_{i=1}^{j} q_i\right)$.

Example 2: Inverse–Gaussian distribution

The inverse–Gaussian distribution, $\mathrm{IG}(\gamma, \eta)$, has a density function given by

$$\pi(\lambda) = \frac{\gamma}{\sqrt{2\pi}} \lambda^{-3/2} \exp\left\{ -\frac{1}{2}\left(\frac{\gamma^2}{\lambda} + \eta^2 \lambda \right) + \eta\gamma \right\},$$

where γ and η can be interpreted as a shape and a scale parameter, respectively. We take λ_j to follow independent $\mathrm{IG}(\gamma_j, 1)$ distributions. Then $\Lambda_m = \sum_{j=m+1}^{\infty} \lambda_j$ is dis-

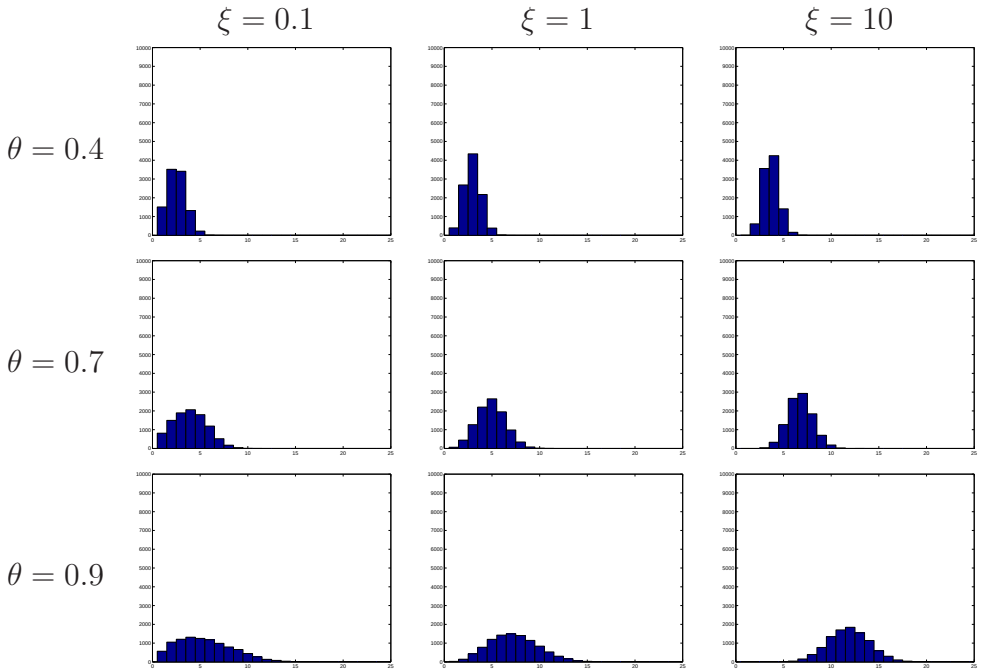

Figure 2: Prior distribution of the number of clusters for infinite normalized inverse–Gaussian prior

tributed as $\mathrm{IG}(\sum_{j=m+1}^{\infty} \gamma_j, 1)$ and the normalization is well–defined if $\sum_{j=1}^{\infty} \gamma_j < +\infty$ which implies that Λ is almost surely finite. The finite dimensional normalized distribution $(\lambda_1/\Lambda, \lambda_2/\Lambda, \ldots, \lambda_m/\Lambda)$ has been studied by Lijoi *et al.* (2005) as the normalized inverse–Gaussian distribution. We again define $\gamma_j = \xi q_j$ and it follows directly from their results that w_i has expectation q_i and variance $q_i(1-q_i)\xi^2 \exp\{\xi\}\Gamma(-2, \xi)$. This prior will be referred to as the infinite normalized inverse–Gaussian prior. Figure 2 shows the prior distribution of the number of clusters in 30 observations. The effects of ξ and θ follow the same pattern as the infinite Dirichlet case discussed above. However, the effect of ξ is less marked for small ξ. In the infinite Dirichlet case for $\xi = 0.1$, the distributions are almost indistinguishable for different values of θ but in

this case it is clear that the location of the distribution is increasing with θ. This allows easier prior specification for the infinite normalized–inverse Gaussian prior

3.2 Slice sampler

The model can be fitted using an extension of the slice sampler developed in section 2. We will assume that the distribution of Λ_m has a known form for all m, which we will denote by $\pi^\star_m(\Lambda_m)$. The introduction of a normalizing constant, Λ, makes MCMC trickier. Simpler updating is possible when we introduce the additional latent variable v, and consider the joint density

$$f(y, v, u, d) = \exp(-v\Lambda)\, \mathbf{1}(u < \lambda_d)\, K(y; \phi_d).$$

Clearly the marginal density

$$f(y, d) = \frac{\lambda_d}{\Lambda} K(y; \phi_d),$$

as required. The likelihood function based on a sample of size n is given by

$$\prod_{i=1}^{n} \exp(-v_i\Lambda)\, \mathbf{1}(u_i < \lambda_{d_i})\, K(y_i; \phi_{d_i}).$$

We will only consider those conditional distributions which are not immediately trivial; those that are completely trivial being u_i, v_i and ϕ_j. The distribution of d_i is trivial but as before we need to find the number of λ_j's (and also ϕ_j's) to be sampled in order to implement the sampling of d_i.

Hence, the non–trivial aspect to the algorithm is the sampling of the sufficient number of $\{\lambda_j\}$ and Λ. We will, as before, work on the conditional distribution of the $(\{\lambda_j\}, \Lambda)$ excluding the $\{u_i\}$. We simulate $\lambda_1, \ldots, \lambda_m, \Lambda_m$ (where m is the number of atoms given in the previous iteration) in a block from their full conditional distribution which is proportional to

$$\exp\{-V\Lambda_m\}\pi^\star_m(\Lambda_m) \prod_{j=1}^{m} \exp\{-V\lambda_j\}\lambda_j^{n_j}\pi_j(\lambda_j),$$

where $n_j = \sum_{i=1}^{n} \mathbf{1}(d_i = j)$ and $V = \sum_{i=1}^{n} v_i$. We need to find the smallest value of m' for which $\Lambda_{m'} < \min_i\{u_i\}$ so that we can evaluate the full conditional distribution of d_i. This value can be found by sequentially simulating $[\lambda_j, \Lambda_j|\Lambda_{j-1}]$ for $j = m + 1, \ldots, m'$. The conditional distribution of $[\lambda_j = x, \Lambda_j = \Lambda_{j-1} - x|\Lambda_{j-1}]$ is given by

$$f(x) \propto \pi_j(x)\pi^\star_j(\Lambda_{j-1} - x), \qquad 0 < x < \Lambda_{j-1}.$$

In some cases simulation from the distribution will be straightforward. If not, generic univariate simulation methods such as Adaptive Rejection Metropolis Sampling (Gilks *et al.* 1995) can be employed. We now consider a couple of examples.

Example 1: Gamma distribution
It is easy to see that

$$\pi(\lambda_1/\Lambda, \ldots, \lambda_m/\Lambda|\Lambda, \cdots, \text{ exclude } u) = \text{Dir}\left(\gamma_1 + n_1, \ldots, \gamma_m + n_m, \sum_{l=m+1}^{\infty} \gamma_l\right)$$

and

$$\Lambda_m \sim \text{Ga}\left(\sum_{j=m+1}^{\infty} \gamma_j, 1 + V\right).$$

The conditional distribution of λ_j/Λ_j is $\text{Be}(\gamma_j, \sum_{i=j+1}^{\infty} \gamma_i)$. This prior can also be represented as a stick–breaking prior.

Example 2: Inverse–Gaussian distribution
The full conditional distribution of λ_j is given by

$$\pi(\lambda_j|\cdots) \propto \lambda_j^{n_j - 3/2} \exp\left\{-\frac{1}{2}\left(\frac{\gamma_j^2}{\lambda_j} + (1 + 2V)\lambda_j\right)\right\},$$

where n_j is the number of observations allocated to component j. The full conditional distribution of Λ_m is proportional to

$$\Lambda_m^{-3/2} \exp\left\{-\frac{1}{2}\left(\frac{(\sum_{i=m+1}^{\infty} \gamma_i)^2}{\lambda_j} + (1 + 2V)\lambda_j\right)\right\}.$$

These are both generalized inverse–Gaussian distributions which can be simulated directly; see e.g. Devroye (1986).

We can simulate from $[\lambda_{j+1}, \Lambda_{j+1}|\Lambda_j]$ by defining $\lambda_{j+1} = x_{j+1}\Lambda_j$ and $\Lambda_{j+1} = (1 - x_{j+1})\Lambda_j$ where the density of x_{j+1} is given by

$$g(x_{j+1}) \propto x_{j+1}^{-3/2}(1 - x_{j+1})^{-3/2} \exp\left\{-\frac{1}{2}\left[\frac{\gamma_j^2}{\Lambda_j x_{j+1}} + \frac{(\sum_{i=j+1}^{\infty} \gamma_i)^2}{\Lambda_j(1 - x_{j+1})}\right]\right\}.$$

Unlike the gamma case, this conditional distribution depends on Λ_m. The distribution of $x_{j+1}/(1 - x_{j+1})$ can be identified as a two–mixture of generalized inverse–Gaussian distributions and hence can be sampled easily (details are given in the Appendix).

4 Hazard Functions

The normalized procedure can also be applied to the modeling of random hazard functions. Suppose we model the unknown hazard function $h(t)$, for $t > 0$, using a set of known functions $\{h_k(t)\}_{k=1}^{\infty}$, via

$$h(t) = \sum_{k=1}^{\infty} \lambda_k \, h_k(t).$$

Here the $\{\lambda_k > 0\}$ are the model parameters and can be assigned independent gamma prior distributions; say $\lambda_k \sim \text{Ga}(a_k, b_k)$. Obviously we will need to select (a_k, b_k) to ensure that $h(t) < +\infty$ a.s. for all $t < +\infty$. The corresponding density function is given by

$$f(t) = \sum_{k=1}^{\infty} \lambda_k h_k(t) \, \exp\left\{ - \sum_{k=1}^{\infty} \lambda_k H_k(t) \right\},$$

where H_k is the cumulative hazard corresponding to h_k.

So with observations $\{t_i\}_{i=1}^{n}$, the likelihood function is given by

$$l(\lambda | t) \propto \prod_{i=1}^{n} \left[\sum_{k=1}^{\infty} \lambda_k h_k(t_i) \, \exp\left\{ - \sum_{k=1}^{\infty} \lambda_k H_k(t_i) \right\} \right].$$

Our approach is based on the introduction of a latent variable, say u, so that we consider the joint density with t given by

$$f(t, u) = \sum_{k=1}^{\infty} \mathbf{1}(u < \lambda_k) \, h_k(t) \, \exp\left\{ - \sum_{k=1}^{\infty} \lambda_k H_k(t) \right\}.$$

A further latent variable d picks out the mixture component from which (t, u) come,

$$f(t, u, d) = \mathbf{1}(u < \lambda_d) \, h_d(t) \, \exp\left\{ - \sum_{k=1}^{\infty} \lambda_k H_k(t) \right\}.$$

We will now introduce the key latent variables, one for each observation, and label them (u_i, d_i), into the likelihood, which is given by

$$l(\lambda | t, u, d) \propto \prod_{i=1}^{n} \mathbf{1}(u_i < \lambda_{d_i}) \, h_{d_i}(t_i) \, \exp\left\{ - \sum_{k=1}^{\infty} \lambda_k H_k(t_i) \right\}.$$

The point is that the choice of d_i is finite. It is now clear that the sampling algorithm for this model is basically the same now as for the normalized case. We could take

the λ_j to be gamma with parameters $a_j + \sum_{d_i=j} 1$ and $b_j + \sum_{d_i=j} H_j(t_i)$ and we would first sample up to $M = \max_i d_i$. Then the u_i are from $\mathrm{Un}(0, \lambda_{d_i})$. In order to sample the d_i we need to find all the λ_j greater than u_i. We can do this by sampling $\Lambda_M = \sum_{j>M} \lambda_j$ as a gamma distribution and then sampling $[\lambda_{M+1}, \ldots \lambda_{N_i}]|\Lambda_M$ so that N_i is the smallest integer for which $\sum_{j=M+1}^{N_i} \lambda_j > \Lambda_M - u_i$. Finally, once we have found all the $\lambda_j > u_i$, we can sample d_i from $\Pr(d_i = j) \propto \mathbf{1}(\lambda_j > u_i) h_j(t_i)$.

5 Illustration and Comparisons

In this section we carry out a comparison of the slice sampling algorithm with the retrospective sampler using the Dirichlet process and the normalized weights prior. The algorithms are compared using the normal kernel $K(y|\phi)$ with components $\phi = (\mu, \zeta)$, and $P_0(\mu, \zeta) = \mathrm{N}(\mu|\mu_0, \xi^2) \times \mathrm{G}(\zeta|\gamma, \beta)$. Here $\mathrm{G}(\gamma, \beta)$ denotes the gamma distribution. We also consider inference for the commonly used galaxy data set with the infinite Dirichlet and infinite normalized inverse–Gaussian priors.

For comparison purposes we consider two real data sets and two simulated data sets. The real data sets are:

1. Galaxy data set which consists of the velocities of 82 distant galaxies diverging from our own galaxy. This is the most commonly used data set in density estimation studies, due to its mulimodality. We will also use it to illustrate the effect of the prior choice on the posterior density in Section 5.3.

2. S & P 500 data set which consist of 2023 daily index returns. This is yet another commonly used data set in density estimation and volatility studies of financial asset returns; see, Jacquier, Polson, and Rossi (1994, 2004). This data set is unimodal, not necessarily symmetric, around zero, and it is characterized by heavy tails.

We chose these data sets because of their size, as we would like to study the performance of the algorithms on both small and large data sets.

The simulated data sets are based on the models used in Green and Richardson (2001) and consist of 100 draws from a bimodal and a leptokurtic mixture.

1. The bimodal mixture: $0.5N(-1, 0.5^2) + 0.5N(1, 0.5^2)$.

2. The leptokurtic mixture: $0.67N(0, 1) + 0.33N(0.3, 0.25^2)$.

Both of these simulated data sets were used in the algorithm comparison study carried out in Papaspiliopoulos and Roberts (2008); since we are comparing our slice sampler with the retrospective sampler, we decided to use these simulated data sets.

The parameters for our MDP mixture are also set according to Green and Richardson (2001). If R is the range of the data; then we take $\mu_0 = R/2$, $\xi = R$, $\gamma = 2$, and $\beta = 0.2R^2$. The precision parameter of the Dirichlet Process is set at $M = 1$. In the comparison of the estimates of the statistics used, we took the Monte Carlo sample size to be $S = 250,000$ for each algorithm, with the initial $10,000$ used as a burn in period. Density estimates using the retrospective and slice–efficient samplers are shown in figure 3 for the Dirichlet process mixture model.

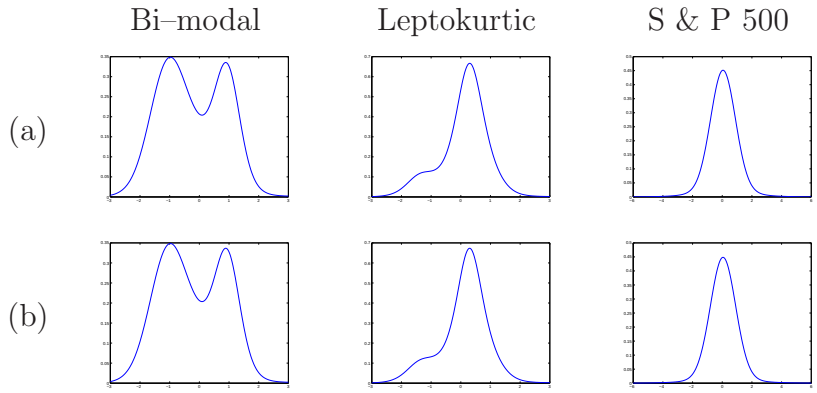

Figure 3: Predictive densities: (a) retrospective and (b) slice–efficient

5.1 Algorithmic performance

To monitor the performance of the algorithms we look at the convergence of two quantities:

- The number of clusters: at each iteration there are $j = 1, \ldots, N$ clusters of the $i = 1, \ldots, n$ data points with m_j being the size of the j cluster, so that $\sum_{j=1}^{N} m_j = n$.
- The deviance, D, of the estimated density, calculated as

$$D = -2 \sum_{i=1}^{n} \log \left(\sum_{j} \frac{m_j}{n} K(y_i | \phi_j) \right).$$

These variables have been used in the previous comparison studies of Papaspiliopoulos and Roberts (2008), Green and Richardson (2001) and Neal (2000). Here D is one of the most common functionals used in comparing algorithms, because it is seen as a global function of all model parameters. Although we produce this variable and study its algorithmic performance we are also concerned with the convergence of the number of clusters.

The efficiency of the algorithms is summarized by computing an estimate $\widehat{\tau}$ of the integrated autocorrelation time, τ, for each of the variables. Integrated autocorrelation time is defined in Sokal (1997) as

$$\tau = \frac{1}{2} + \sum_{l=1}^{\infty} \rho_l.$$

where ρ_l is the autocorrelation at lag l. An estimate of τ has been used in Papaspiliopoulos and Roberts (2008), Green and Richardson (2001) and Neal (2000). Integrated autocorrelation time is of interest as it controls the statistical error in Monte Carlo measurements of a desired function f. To clarify this point, consider the Monte Carlo sample mean,

$$\bar{f} \equiv \frac{1}{S} \sum_{l=1}^{S} f_j,$$

where S is the number of iterations. The variance of $\bar{f}$ according to Sokal (1997) is

$$\mathrm{Var}(\bar{f}) \approx \frac{1}{S} 2\tau \times V,$$

where V is the marginal variance. Sokal (1997) concludes that $\mathrm{Var}(\bar{f})$ is a factor 2τ larger than what it would be if the $\{f_j\}$ were statistically independent. In other words, τ determines the statistical error of the Monte Carlo measurements of f once equilibrium has been attained. Therefore a run of S iterations contains only $S/(2\tau)$ *"effectively independent data points"*. This means that the algorithm with the smallest estimated value of τ will be the most efficient. The problem with the calculation of τ lies in accurately estimating the covariance between the states, which in turn is used to calculate the autocorrelation ρ_l. It must be noted that in MCMC the covariance and the autocorrelation are not single values but random variables. Based on Sokal (1997) the estimator for τ is given by

$$\widehat{\tau} = \frac{1}{2} + \sum_{l=1}^{C-1} \widehat{\rho}_l.$$

where $\widehat{\rho}_l$ is the estimated autocorrelation at lag l (obtained via MatLab) and C is a cut–off point, normally set by the researcher. In our comparisons we define, as is commonly done,

$$C = \min\left\{l : |\widehat{\rho}_l| < 2/\sqrt{S}\right\}.$$

Then C is the smallest lag for which we would not reject the null hypothesis $H_0 : \rho_l = 0$. A similar approach has also been used in Papaspiliopoulos (2008). The issue here, see Sokal (1997), is the cut off point C; it introduces bias equal to

$$\text{Bias}(\widehat{\tau}) = \frac{1}{2}\sum_{|l|>C}\rho_l + o\left(\frac{1}{S}\right).$$

On the other hand, the variance of $\widehat{\tau}$ can be computed using

$$\text{Var}(\widehat{\tau}) \approx \frac{2(2C-1)}{S}\tau^2.$$

The choice of C will be a trade off between the bias and the variance of $\hat{\tau}$, which means that we really cannot say how "good" an algorithm is since the choice of C point is left to the researcher. According to Sokal (1997), this approach works well when a sufficient quantity of data is available which we can control by running the sampler for a sufficient number of iterations.

5.2 Results

The following tables compare the estimated integrated autocorrelation time $\widehat{\tau}$ of the two variables of interest; the number of clusters and the deviance.

5.2.1 Dirichlet process

Looking at the estimates of $\widehat{\tau}$ for the real data sets we come to the following conclusions:

- For the galaxy data set there is little difference between the two samplers. Even though the retrospective sampler performs marginally better, the slice–efficient sampler is easier to use as simulating the z and k is carried out in an easy way, as opposed to the complexity of the set up of the retrospective sampling steps.
- For the S&P data set which is large, unimodal, asymmetric and heavy–tailed, it is the slice–efficient sampler that outperforms the retrospective sampler, in terms of $\widehat{\tau}$ for the number of clusters; $\widehat{\tau}$ for the slice–efficient sampler is about half that of the retrospective sampler.

	Galaxy data		Leptokurtic data	
	$\widehat{\tau}$ for $\sharp$ clust	$\widehat{\tau}$ for D	$\widehat{\tau}$ for $\sharp$ clust	$\widehat{\tau}$ for D
Slice	31.5268	10.2683	157.0064	119.3368
Slice–efficient	10.2868	4.3849	33.0470	26.0547
Retrospective	6.7677	2.9857	13.6639	9.3014

	Bimodal data		S&P 500 data	
	$\widehat{\tau}$ for $\sharp$ clust	$\widehat{\tau}$ for D	$\widehat{\tau}$ for $\sharp$ clust	$\widehat{\tau}$ for D
Slice	167.4995	54.6059	142.6566	81.4236
Slice–efficient	26.8114	10.8374	4.1923	5.2390
Retrospective	14.7202	7.1603	7.1464	1.5779

Table 1: Estimates of the integrated autocorrelation times for the deviance (D) and for the number of clusters with four data sets with the Dirichlet process mixture model

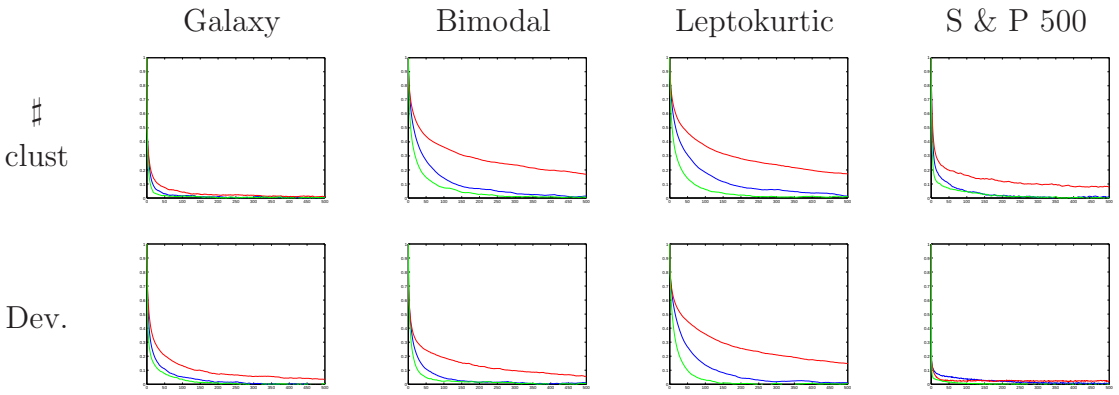

Figure 4: Autocorrelation of MCMC output for: slice sampler (red), efficient slice sampler (blue) and retrospective sampler (green)

5.2.2 Mixtures based on normalized weights

We reject the slice sampler in favour of the slice–efficient sampler. We use the infinite Dirichlet and infinite normalized inverse–Gaussian mixtures models with $\xi = 1$ and $\theta = 0.5$ on the four data sets. We find similar performance for the normalized weights prior as for the Dirichlet process prior. The retrospective sampler is usually more efficient than the slice sampler with a two times relative improvement on average.

	Galaxy data		Leptokurtic data	
	$\widehat{\tau}$ for $\sharp$ clust	$\widehat{\tau}$ for D	$\widehat{\tau}$ for $\sharp$ clust	$\widehat{\tau}$ for D
Slice–efficient	25.50	12.21	115.36	79.70
Retrospective	27.12	7.08	48.32	29.13

	Bimodal data		S&P 500 data	
	$\widehat{\tau}$ for $\sharp$ clust	$\widehat{\tau}$ for D	$\widehat{\tau}$ for $\sharp$ clust	$\widehat{\tau}$ for D
Slice–efficient	64.19	17.03	21.69	11.99
Retrospective	44.05	8.64	14.17	3.22

Table 2: Estimates of the integrated autocorrelation times for the deviance (D) and for the number of clusters with four data sets with the infinite Dirichlet distribution mixture model

The improvement is typically larger for the simulated rather than the real data sets. The effect is also more pronounced for the infinite Dirichlet distribution prior rather than the infinite normalized inverse–Gaussian prior.

	Galaxy data		Leptokurtic data	
	$\widehat{\tau}$ for $\sharp$ clust	$\widehat{\tau}$ for D	$\widehat{\tau}$ for $\sharp$ clust	$\widehat{\tau}$ for D
Slice–efficient	22.41	8.89	41.95	31.64
Retrospective	16.91	4.75	27.63	21.52

	Bimodal data		S&P 500 data	
	$\widehat{\tau}$ for $\sharp$ clust	$\widehat{\tau}$ for D	$\widehat{\tau}$ for $\sharp$ clust	$\widehat{\tau}$ for D
Slice–efficient	34.72	15.79	28.42	8.38
Retrospective	23.20	9.45	85.57	3.01

Table 3: Estimates of the integrated autocorrelation times for the deviance (D) and for the number of clusters with four data sets with the infinite normalized inverse–Gaussian distribution mixture model

These results are not surprising. The slice sampler introduces auxiliary variables to help simulation which will slow convergence through over–conditioning. The slice–efficient sampler reduces this effect by jointly updating u and λ (or V in the Dirichlet

process case) in a block. The retrospective sampler will mix slowly when the proposal distribution is a poor approximation to the full conditional distribution. Therefore it is usually difficult to be sure about the ranking of the methods. In these illustrations, we have seen examples where the slice–efficient sampler is more efficient than the retrospective sampler.

5.3 Inference for the Normalized Weights Priors

The galaxy data has been a popular data set in Bayesian nonparametric modelling and we will illustrate the infinite Dirichlet and infinite normalized inverse–Gaussian priors on it. The posterior mean density estimates are shown in figure 5 for the infinite Dirichlet prior and figure 6 for the infinite normalized inverse–Gaussian prior. The hyperparameters of the prior distributions have a clear effect on the posterior

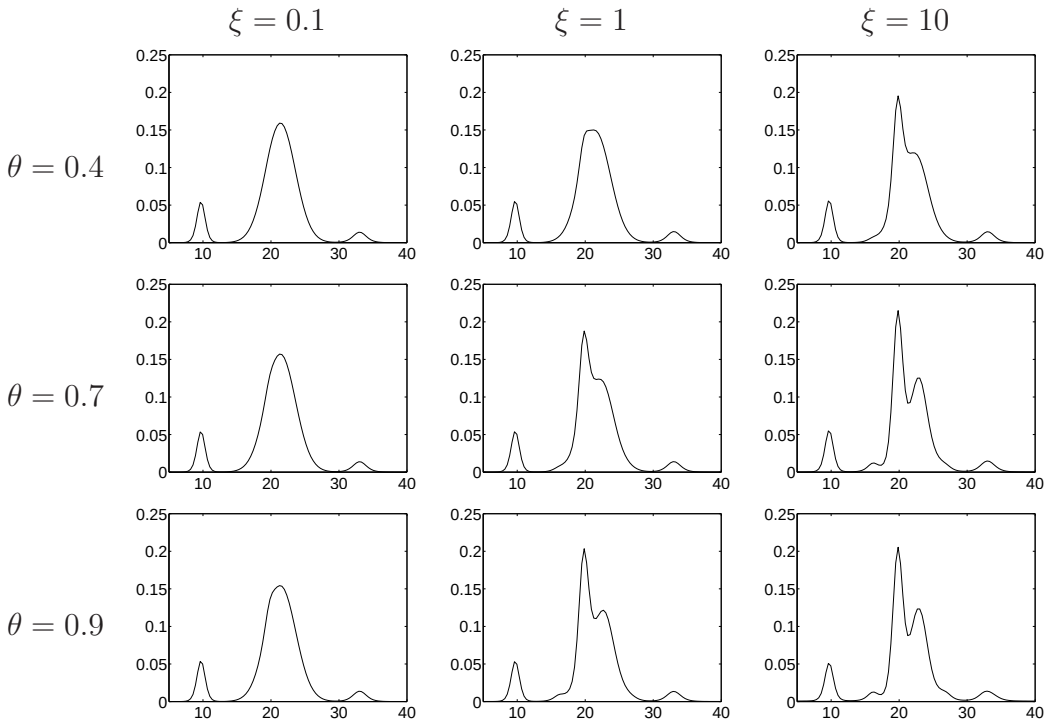

Figure 5: Posterior mean density estimates for the galaxy data using the infinite Dirichlet prior with different values of M and θ

mean estimates. Prior distributions that places more mass on a small number of components tend to find estimates with three clear modes. As the prior mean number of components increases so do the number of modes in the estimate from 4 to 5 for

the prior within each class that places most mass on a large number of components ($\xi = 10$ and $\phi = 0.9$). However, there are some clear differences between the two

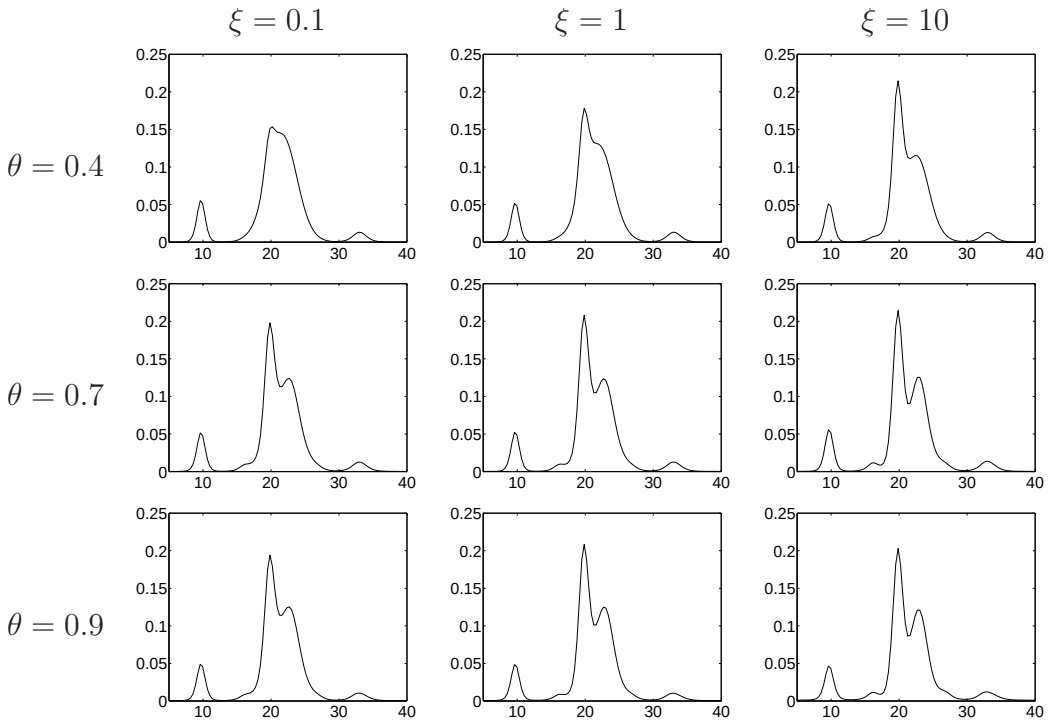

Figure 6: Posterior mean density estimates for the galaxy data using the infinite normalized inverse–Gaussian prior with different values of ξ and θ

classes of prior. The effects of the two hyperparameters on the prior distribution of the number of non–empty components were more clearly distinguishable in the infinite normalized inverse–Gaussian prior than the infinite Dirichlet prior. In the infinite normalized inverse–Gaussian prior θ controls the mean number of non–empty components whereas ξ controls the dispersion around the mean. This property is carried forward to the posterior mean density and the number of modes in the posterior mean increases with θ. For example, when $\xi = 0.1$, there are three modes in the posterior mean if $\theta = 0.4$ whereas there are 4 when $\theta = 0.9$. Similarly, larger values of ξ are associated with larger variability in the prior mean and favour distributions which uses a larger number of components. This suggests that infinite normalized inverse–Gaussian distribution may be a more easily specified prior distribution than the infinite Dirichlet prior.

6 Conclusions and Discussion

This paper has shown how mixture models based on random probability measures, of either the stick–breaking or normalized types, can be easily handled via the introduction of a key latent variable which makes finite the number of mixtures. The more complicated of the two is the normalized type, which requires particular distributions of the unnormalized weights in order to be able to make the simulation algorithm work. Nevertheless, such distributions based on the gamma and inverse–Gaussian distributions are popular choices anyway.

Further ideas which need to be worked out include the case when we can generate weights which are decreasing. This for example would make the search for those $w_j > u$ are far simpler exercise and would lead to more efficient algorithms.

In conclusion, concerning performance of slice–efficient and retrospective samplers, we note that once running, both samplers are approximately the same in terms of efficiency and performance. In terms of time efficiency we have found that for large data sets, like the S&P 500 the slice–efficient sampler is more efficient than the retrospective sampler, it takes approximately half the time to run than the retrospective sampler. The most notable savings of the slice–efficient sampler are in the pre–running work where setting up a slice sampler is far easier than setting up a retrospective sampler.

The slice sampler allows the Gibbs sampling step for a finite mixture model to be used at each iteration and introduce a method for updating the truncation point in each iteration. This allows standard methods for finite mixture models to be used directly. For example, Van Gael *et al* (2008) fit an infinite hidden Markov model using the forward–backward sampler for finite hidden Markov model using the slice sampling idea. This would be difficult to implement in a retrospective framework since the truncation point changes when updating the allocations.

References

Devroye, L. (1986): "Non–Uniform Random Variate Generation," Springer–Verlag: New York.

Escobar, M.D. 1988. Estimating the means of several normal populations by nonparametric estimation of the distribution of the means. Unpublished Ph.D. dissertation, Department of Statistics, Yale University.

Escobar, M.D. (1994). Estimating normal means with a Dirichlet process prior. *Journal of the American Statistical Association* **89**, 268–277.

Escobar, M.D. and West, M. (1995). Bayesian density estimation and inference using mixtures. *Journal of the American Statistical Association* **90**, 577–588.

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

Gilks, W.R., Best, N.G. and Tan, K.K.C. (1995). Adaptive rejection Metropolis sampling within Gibbs sampling. *Applied Statistics* **44**, 455–472.

Green, P. J. and Richardson, S. (2001). Modelling heterogeneity with and without the Dirichlet process. *Scandinavian Journal of Statistics* **28**, 355–375.

Ishwaran, H. and James, L.F. (2001). Gibbs sampling methods for stick–breaking priors. *Journal of the American Statistical Association* **96**, 161–173.

Jacquier, E., Polson, N., and Rossi P.E. (1994). Bayesian Analysis of stochastic volatility models. *Journal of Business and Economic Statistics* **12**, 371–417.

Jacquier, E., Polson, N., and Rossi P.E. (2004). Bayesian Analysis of stochastic volatility models with fat tails and correlated errors. *Journal of Econometrics* **122**, 185–212.

Lijoi, A., Mena, R. H. and Prünster, I. (2005). Hierarchical Mixture Modeling with Normalized Inverse–Gaussian Priors. *Journal of the American Statistical Association* **100**, 1278–1291.

Lijoi, A., Mena, R. H. and Prüenster, I (2007): "Controlling the reinforcement in Bayesian nonparametric mixture models," *Journal of the Royal Statistical Society B*, **69**, 715–740.

Lo, A.Y. (1984). On a class of Bayesian nonparametric estimates I. Density estimates. *Annals of Statistics* **12**, 351–357.

MacEachern, S.N. (1994). Estimating normal means with a conjugate style Dirichlet process prior. *Communications in Statistics: Simulation and Computation* **23**, 727–741.

MacEachern, S.N. and Müller, P. (1998). Estimating mixtures of Dirichlet process models. *Journal of Computational and Graphical Statistics* **7**, 223–238.

Neal, R. (2000). Markov chain sampling methods for Dirichlet process mixture models. *Journal of Computational and Graphical Statistics* **9**, 249–265.

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

Papaspiliopoulos, O. (2008). A note on posterior sampling from Dirichlet mixture models. Preprint.

Sethuraman, J. (1994). A constructive definition of Dirichlet priors. *Statistica Sinica* **4**, 639–650.

Sokal, A. (1997). Monte Carlo Methods in Statistical Mechanics: Foundations and New Algorithms. *Functional Integration (Cargése, 1996)* **361***of NATO Adv. Sci. Inst. Ser. B Phys.*, New York: Plenum, 131–192.

Smith, A.F.M. and Roberts, G.O. (1993). Bayesian computations via the Gibbs sampler and related Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society, Series B* **55**, 3–23.

Van Gael, J., Saatchi, Y., Teh, Y.W., and Ghahramani, Z. (2008). Beam Sampling for the Infinite Hidden Markov Model. *Technical Report : Engineering Department, University of Cambridge*.

Walker, S.G. (2007). Sampling the Dirichlet mixture model with slices. *Communications in Statistics: Simulation and Computation* **36**, 45–54.

Appendix

Simulation for the Inverse–Gaussian model. We wish to simulate from the density $g(x_{j+1})$

$$g(x_{j+1}) \propto x_{j+1}^{-3/2}(1-x_{j+1})^{-3/2} \exp\left\{-\frac{1}{2}\left[\frac{\gamma_j^2}{\Lambda_j x_{j+1}} + \frac{(\sum_{i=j+1}^{\infty}\gamma_i)^2}{\Lambda_j(1-x_{j+1})}\right]\right\}.$$

The transformation $y_{j+1} = \frac{x_{j+1}}{1-x_{j+1}}$ has the density

$$g(y_{j+1}) \propto y_{j+1}^{-3/2}(1+y_{j+1}) \exp\left\{-\frac{1}{2}\left[\frac{\gamma_j^2}{\Lambda_j y_{j+1}} + \frac{(\sum_{i=j+1}^{\infty}\gamma_i)^2}{\Lambda_j} y_{j+1}\right]\right\}.$$

which can be expressed as a mixture of two generalized inverse–Gaussian distributions

$$w\, GIG\left(-1/2, \gamma_j/\Lambda_j, \sum_{i=j+1}^{\infty}\gamma_i/\Lambda_j\right) + (1-w)\, GIG\left(1/2, \gamma_j/\Lambda_j, \sum_{i=j+1}^{\infty}\gamma_i/\Lambda_j\right)$$

where

$$w = \frac{\gamma_j}{\sum_{i=j+1}^{\infty}\gamma_i}$$

and $GIG(p,a,b)$ denotes a distribution with density

$$\frac{(b/a)^{p/2}}{2K_p\left(\sqrt{ab}\right)} x^{(p-1)} \exp\{-(a/x+bx)/2\}$$

where K_ν denotes the modified Bessel function of the third kind with index ν.