Sequential Monte Carlo Methods for Normalized Random Measure with Independent Increments Mixtures

J.E. Griffin*
School of Mathematics, Statistics and Actuarial Science,
University of Kent, Canterbury, CT2 7NF, U.K.

June 10, 2011

Abstract

Normalized random measures with independent increments are a tractable and wide class of nonparametric prior. Sequential Monte Carlo methods are developed for both conjugate and non-conjugate models. Methods for improving efficiency by including Markov chain Monte Carlo steps without increasing computational complexity are discussed. A simulation study is used to compare the efficiency of the different algorithms for density estimation. The methods are further illustrated by application to estimation of the marginal likelihood in a goodness-of-fit testing example and clustering of time series using a non-conjugate mixture model.

keywords Particle filtering; Bayesian nonparametrics; Dirichlet process; Normalized generalized Gamma process; Clustering time series; Slice sampling

*Corresponding author: Jim Griffin, School of Mathematics, Statistics and Actuarial Science, University of Kent, Canterbury, CT2 7NF, U.K. Tel.: +44-1227-; Fax: +44-1227-; Email: J.E.Griffin-28@kent.ac.uk.

1 Introduction

The infinite mixture model has become a popular method for Bayesian nonparametric density estimation and clustering. It is assumed that a random sample $y_1, \ldots, y_n$ are independent and that the density of y_t is

$$f(y_t) = \sum_{k=1}^{\infty} w_k k(y_t \mid \theta_k, \phi) \qquad (t = 1, \ldots, n) \tag{1}$$

where $k(x \mid \theta, \phi)$ is a probability density function for x with parameters θ and ϕ, $w_k > 0 \quad (k = 1, 2, \ldots)$ and $\sum_{k=1}^{\infty} w_k = 1$, and $\theta_1, \theta_2, \cdots \overset{i.i.d.}{\sim} H$ (whose density is h if H is continuous). The most popular instance of this model is the Dirichlet process mixture (Escobar and West, 1995) where $w_1, w_2, \ldots$ are derived from the Dirichlet process. This prior is computationally attractive but the choice can restrict the forms of clustering available and so tractable generalizations have been proposed. Ishwaran and James (2001) describe the construction of stick-breaking priors and James et al. (2009) discuss inference in the class of normalized random measures with independent increments (NRMI). This paper will concentrate on mixtures using the latter class of priors and describe sequential Monte Carlo methods for inference.

Constructing a method for posterior inference is challenging in infinite mixture models since an infinite number of parameters is involved and the posterior is typically intractable. Various Markov chain Monte Carlo methods have been developed which represent the posterior in terms of finite-dimensional objects that can be sampled. In Dirichlet process mixture models, Gibbs sampling methods for conjugate model, where $\int h(\theta) \prod_{j=1}^{m} k(y_{t_j} \mid \theta, \phi)\, d\theta$ for any subset $\{t_1, \ldots, t_m\}$ of $\{1, \ldots, n\}$ can be calculated analytically, were proposed by Escobar and West (1995) and for non-conjugate model by MacEachern and Müller (1998) and Neal (2000). These methods effectively integrate over the infinite dimensional prior using the Pólya urn scheme representation of the Dirichlet process to define a finite-dimensional posterior. The conjugate approach can be directly extended to NRMI mixtures using the Pólya urn scheme derived by James et al. (2009) but there has been no work extending the methods of Neal (2000) to this wider class of priors.

Methods which use the Pólya urn scheme representation are often called marginal methods. Alternatively, conditional methods include the random measure in the sampler and are more suitable for non-conjugate models. A first step in this direction was taken by Ishwaran and James (2001) where a finite approximation of a stick-breaking process is derived which controls the truncation error of functions of the posterior distribution and

leads to an efficient blocked Gibbs sampling scheme. Subsequently, it was discovered that the truncation error can be completely removed using truncations of random length. Two such Markov chain Monte Carlo methods for stick-breaking processes are retropsective sampling (Papaspiliopoulos and Roberts, 2008) and slice sampling (Walker, 2007; Kalli et al., 2011). Griffin and Walker (2011) describe slice sampling methods for non-conjugate NRMI mixtures and these methods will be extended to sequential Monte Carlo sampling in this article.

Sequential Monte Carlo methods build an approximation of the posterior conditional on $y_1, \ldots, y_t$ from the approximation conditional on $y_1, \ldots, y_{t-1}$. Repeated application of this process leads to the posterior conditional on $y_1, \ldots, y_n$. Their application to Dirichlet process mixture models was initially developed by Liu (1996) and MacEachern et al. (1999). They described sequential importance sampling methods which exploited the Pólya urn scheme representation of the Dirichlet process and involved expensive numerical integrations for non-conjugate models. In practice, these algorithms can often perform poorly and lead to estimates with large variances. Fearnhead (2004) extends their algorithm to Sampling-Importance-Resampling algorithm (also known as a particle filter). Chopin (2002) describes the application of a similar algorithm to finite mixture models. There has recently been renewed interested in sequential Monte Carlo methods for infinite mixture models. Ulker et al. (2010) describe elaborations of the the algorithm of Fearnhead (2004) and Carvalho et al. (2010) describe particle learning methods for these models.

Efficient MCMC sampling methods have been an important factor in the popularity of the Dirichlet process mixture model. However, MCMC methods may become stuck in local modes if the posterior has well-separated areas of substantial probability (which may occur in mixture models). The development of more structured infinite mixture models (such as models that allow the component weights to depend on covariates) can potentially exacerbate this problem. Sequential Monte Carlo methods offer an alternative which can potentially avoid these problems. This paper does not directly deal with these problems but is a step in that direction. This paper describes method for both conjugate and non-conjugate models and the estimation of unknown hyperparameters. There has been little work on the latter two inference problem with the exception of Carvalho et al. (2010). The Normalized Random Measures with Independent Increments priors is very large and underlies recently developed time-series and spatial nonparametric priors (Griffin, 2011; Rao and Teh, 2009). Sequential Monte Carlo also allows unbiased estimation of the marginal likelihood which has traditionally been a challenging problem in Bayesian nonparametric inference.

2 Sequential Monte Carlo methods for DP mixture models

Fearnhead (2004) describes a sequential Monte Carlo algorithm for the model in (1) when $w_1, w_2 \dots$ are weights derived from the Dirichlet process. It is convenient to write the model in terms of allocation variables $s_1, \dots, s_n$ which link the observations to the components of the mixture model so that

$$\begin{aligned} y_t &\sim k(y_t \mid \theta_{s_t}, \phi) \qquad (t = 1, \dots, n) \\ p(s_t = k) &= w_k \qquad (t = 1, \dots, n; k = 1, 2, \dots). \end{aligned}$$

The algorithm samples N values $s_{1:t}^{(1)}, \dots, s_{1:t}^{(N)}$ from $p(s_{1:t}|y_{1:t})$ sequentially in t. The notation $x_{i:j} = (x_i, \dots, x_j)$ will be used as shorthand for vectors and $z^{(i)}$ will represent the value of z in the i-th particle. Suppose that there are K_t distinct values of $s_1, \dots, s_t$ and that these are labelled $\{1, \dots, K_t\}$ and let $m_{k,t}$ be the number of $s_j = k$. The details are given in Algorithm 1 which is feasible since pr $\left(y_t \mid s_{1:(t-1)}, s_t = m, y_{1:(t-1)}\right)$ is availabe for conjugate models and pr $\left(s_t = m | s_{1:(t-1)}\right)$ is available from the Pólya urn scheme for the Dirichlet process. This avoids working directly with $w_1, w_2, \dots$ and $\theta_1, \theta_2, \dots$. The algorithm can be very computationally efficient if pr $\left(y_t \mid s_{1:(t-1)}, s_t = m, y_{1:(t-1)}\right)$ can be calculated using sufficient statistics (Fearnhead, 2004).

The algorithm can be extended to non-conjugate mixture models in several ways. Firstly, Algorithm 1 can be directly used if pr $\left(y_t \mid y_{1:(t-1)}, s_{1:(t-1)}, s_t = k\right)$ can be efficiently approximated (using methods such as Monte Carlo integration). This typically restricts us to problems where θ is low-dimensional, often one-dimensional. Secondly, a value $\hat{\theta}^{(i)}_{K^{(i)}_{t-1}+1}$ can be sampled in Step 1a) and pr $\left(y_t \mid y_{1:(t-1)}, s_{1:(t-1)}, s_t = k\right)$ replaced by pr $\left(y_t \mid y_{1:(t-1)}, s_{1:(t-1)}, s_t = k, \hat{\theta}^{(i)}\right)$. This avoids the need to approximate but introduces static parameters into the sequential Monte Carlo sampler which has the associated potential problem of particle degeneracy (where the number of distinct particles is far less than N). Chopin (2002) suggests alleviating this problem by introducing an extra Step 3) where $\hat{\theta}^{(i)}_{1:K^{(i)}_t}$ are updated at the t-th iteration for $i = 1, \dots, N$ using a Markov chain Monte Carlo step such as a Metropolis-Hastings random walk step.

Although, the problem of particle degeneracy for θ is the most serious there is also a problem of particle degeneracy in all sequential Monte Carlo methods for mixture models since $s^{(i)}_{1:t}$ act as static parameters when moving beyond the t-th iteration. Ulker et al. (2010) suggest sampling a block $s_{(t-r):t}$ conditional on $s_{1:(t-r-1)}$ at the t-th iteration to help reduce

For $t = 1, \ldots, n$, perform steps (1) and (2)

1. For $i = 1, \ldots, N$ perform steps (a) and (b)

   (a) Sample $s_t^{(i)}$ conditional on $y_{1:t}$, and $s_{1:(t-1)}^{(i)}$ from

$$q(k) \propto \begin{cases} m_{k,t-1}^{(i)} \, p\left(y_t \mid y_{1:(t-1)}, s_{1:(t-1)}^{(i)}, s_t = k\right) & \text{if } k \leq K_{t-1}^{(i)} \\ M \, p\left(y_t \mid y_{1:(t-1)}, s_{1:(t-1)}^{(i)}, s_t = k\right) & \text{if } k = K_{t-1}^{(i)} + 1 \end{cases}.$$

   (b) Calculate the unnormalized weight

$$\begin{aligned} \psi_t^{(i)} =& M \text{pr}\left(y_t \mid s_{1:(t-1)}^{(i)}, s_t^{(i)} = K_{t-1}^{(i)} + 1, y_{1:(t-1)}\right) \\ &+ \sum_{k=1}^{K_t^{(i)}} m_{k,t-1}^{(i)} \text{pr}\left(y_t \mid s_{1:(t-1)}^{(i)}, s_t^{(i)} = k, y_{1:(t-1)}\right). \end{aligned}$$

2. Re-weight the particles according to the weights $\zeta_i = \frac{\psi_t^{(i)}}{\sum_{i=1}^N \psi_t^{(i)}} \quad (i = 1, \ldots, N)$.

Algorithm 1: SMC algorithm for conjugate DP mixture models

the effect. I will return to this problem in Section 6.

3 Normalized random Measures with independent increment mixtures

Bayesian inference for the Normalized Random Measures with Independent Increment (NRMI) mixtures were discussed by James et al. (2009). Only the class of homogeneous NRMI will be considered which assumes that w_i in (1) are defined by

$$w_k = \frac{J_k}{\sum_{l=1}^{\infty} J_l}$$

where $J_1, J_2, \ldots$ are the jumps of a non-Gaussian Lévy process (*i.e.* a subordinator) with Lévy density $\eta(x)$. The process is well-defined if $0 < \sum_{l=1}^{\infty} J_l < \infty$ which occurs if $\int_0^{\infty} \eta(x)\, dx = \infty$. The choice of $\eta(x)$ controls the rate at which the jumps of the Lévy process decay and this interpretation can be used to define a prior. Several previously proposed priors fit into this class. The Dirichlet process (Ferguson, 1973) with mass parame-

ter M is a normalized Gamma process which has Lévy measure $\eta(x) = Mx^{-1}\exp\{-x\}$ (where $M > 0$). The Normalized Generalized Gamma process (Lijoi et al., 2007) occurs as the normalization of a Generalized Gamma process (Brix, 1999) which has Lévy measure $\eta(x) = \frac{M}{\Gamma(1-\gamma)}x^{-1-\gamma}\exp\{-\lambda x\}$ (where $M > 0$, $0 < \gamma < 1$ and $\theta > 0$). A special case of this class is the Normalized Inverse Gaussian process (Lijoi et al., 2005) which occurs when $\gamma = 1/2$ and $\lambda = 1$.

The joint distribution of the allocations $s_1, \ldots, s_n$ is particularly useful for the conjugate mixture model and can be written

$$\text{pr}(s_1, \ldots, s_t) = E\left[\prod_{k=1}^{K_t} w_k^{m_{k,t}}\right].$$

This is referred to as the Exchangeable Product Partition Formula (EPPF) since it only depends on the values of $s_1, \ldots, s_t$ through $m_{1,t}, \ldots, m_{K_t,t}$. James et al. (2009) use the identity $\int_0^\infty \exp\{-vx\}\, dv = \frac{1}{x}$ to show that this can be conveniently written as

$$\begin{aligned}\text{pr}(s_1, \ldots, s_t) &= \int_0^\infty \cdots \int_0^\infty E\left[\prod_{k=1}^{K_t} J_k^{m_{k,t}} \exp\left\{-\sum_{j=1}^{t} v_j \sum_{l=1}^{\infty} J_l\right\}\right] dv_1 \cdots dv_t \\ &= \int_0^\infty \cdots \int_0^\infty E\left[\prod_{k=1}^{K_t} f(t,t,k)\exp\left\{-L_t\left(v\right)\right\}\right] dv_1 \cdots dv_t \qquad (2)\end{aligned}$$

where

$$f(t,s,k) = \int_0^\infty J_k^{m_{k,t}} \exp\left\{-J_k \sum_{j=1}^{s} v_j\right\} \eta(J_k)\, dJ_k$$

and

$$L_t(v) = \int_0^\infty \left(1 - \exp\left\{-x\sum_{i=1}^{t} v_i\right\}\right) \eta(x)\, dx.$$

Suppose that $G = \sum_{k=1}^{\infty} J_k \delta_{\theta_k}$ then James et al. (2009) prove the following important result. Let $y_1, \ldots, y_t$ be independent and identically distributed according to G then the posterior of G conditional on $v_1, \ldots, v_n$ and $y_1, \ldots, y_t$ is a combination of a finite set of fixed points $(\hat{J}, \hat{\theta})$ where $\hat{\theta}_k$ is equal to the k-th distinct value of $y_1, \ldots, y_t$ and $p(\hat{J}_k \mid y) \propto \eta(\hat{J}_k)\hat{J}_k^{m_{k,t}} \exp\{-\hat{J}_k \sum_{j=1}^{t} v_j\}$ and $(\tilde{J}, \tilde{\theta})$ where $\tilde{J}$ is a Poisson process with intensity $\eta(J)\exp\{-J\sum_{j=1}^{t} v_j\}$ and $\tilde{\theta}_k \overset{i.i.d.}{\sim} H$ (k=1,2,...).

4 Sequential Monte Carlo methods for conjugate NRMI mixtures

Conjugate NRMI mixture models can be fitted by extending the methods for Dirichlet process mixtures described in Section 2. An expression for the conditional distribution of s_t given $s_{1:(t-1)}$ for any NRMI mixture is derived by James et al. (2009). This a finite, discrete distribution but it can be difficult to compute the probabilities of different values of s_t for many choices of $\eta(x)$. An alternative approach which seems to work well in practice is to use the result in (2) and introduce $v_1, \ldots, v_n$ as latent variables, as in Markov chain Monte Carlo algorithm of NRMI mixtures. In this case we need to sample from the joint distribution

$$p(s_t, v_t \mid s_{1:(t-1)}, v_{1:(t-1)}) = p(s_t \mid s_{1:(t-1)}, v_{1:t}) p(v_t \mid s_{1:(t-1)}, v_{1:(t-1)})$$

where

$$p(v_t \mid s_{1:(t-1)}, v_{1:(t-1)}) = \frac{p(s_{1:(t-1)}, v_{1:t})}{p(s_{1:(t-1)}, v_{1:(t-1)})}$$

and

$$p(s_t \mid s_{1:(t-1)}, v_{1:t}) = \frac{p(s_{1:t}, v_{1:t})}{p(s_{1:(t-1)}, v_{1:t})} \tag{3}$$

The following expressions can be derived from (2)

$$p(s_{1:t}, v_{1:t}) = \prod_{k=1}^{K_t} f(t, t, k) \exp\left\{-L_t(v)\right\}$$

and

$$p(s_{1:(t-1)}, v_{1:t}) = -\frac{d}{dv_t} \prod_{k=1}^{K_{t-1}} f(t-1, t, k) \exp\left\{-L_t(v)\right\}.$$

This implies that the density of v_t given $v_{1:(t-1)}, s_{1:(t-1)}$ is proportional to

$$-\frac{d}{dv_t} \prod_{k=1}^{K_{t-1}} f(t-1, t, k) \exp\left\{-L_t(v)\right\}$$

and its distribution function is

$$\frac{\prod_{k=1}^{K_{t-1}} f(t-1, t, k) \exp\left\{-L_t(v)\right\}}{\prod_{k=1}^{K_{t-1}} f(t-1, t-1, k) \exp\left\{-L_t(v)\right\}}.$$

Values of v_t can be simulated either using inversion sampling from the distribution function or, in some cases, using standard methods for sampling from densities. The distribution of s_t is

$$\frac{p(s_{1:(t-1)}, s_t = j, v_{1:t})}{p(s_{1:(t-1)}, v_{1:t})}$$

which is a finite, discrete distribution and so can be sampled easily. The full algorithm for the conjugate mixture model is shown in Algorithm 2

For $t = 1, \ldots, n$, perform steps (1) and (2)

1. For $i = 1, \ldots, N$ perform steps (a)–(c)

   (a) Sample $v_t^{(i)}$ from the distribution $v_t^{(i)} \mid s_{1:(t-1)}^{(i)}, v_{1:(t-1)}^{(i)}$

   (b) Sample $s_t^{(i)}$ from the distribution proportional to

   $$p\left(s_t^{(i)} = k\right) p\left(y_t \mid s_{1:(t-1)}^{(i)}, s_t^{(i)} = k\right).$$

   (c) Calculate the unnormalized weight

   $$\psi_t^{(i)} = \sum_{k=1}^{K_t^{(i)}+1} p\left(s_t^{(i)} = k\right) p\left(y_t \mid s_{1:(t-1)}^{(i)}, s_t^{(i)} = k\right).$$

2. Re-weight the particles according to the weights $\zeta_i = \frac{\psi_t^{(i)}}{\sum_{i=1}^N \psi_t^{(i)}} \quad (i = 1, \ldots, N)$.

Algorithm 2: SMC algorithm for conjugate NRMI mixture models

5 Sequential Monte Carlo methods for non-conjugate NRMI mixtures

Most sequential Monte Carlo methods have concentrated on conjugate models since $\theta_1, \theta_2, \ldots$ and $w_1, w_2, \ldots$ can both be integrated from the model which leaves an algorithm that works directly on the allocation variables $s_1, \ldots, s_n$. Non-conjugate models are typically harder since the random measure cannot be analytically integrated from the model. The methods described in this paper exploit the conjugacy of the jumps which are conditionally independent of θ (which is non-conjugate) and avoid the problem an infinite number of jumps using slice sampling ideas.

A naive (but practically impossible) implementation of a Gibbs sampler for the infinite mixture model would lead to a infinite number of possible values for s_t. Slice sampling methods for infinite mixture models introduce latent variables which make all steps of a Gibbs sampler have finite support. Griffin and Walker (2011) show how such a Gibbs sampler can be efficiently constructed for the class of mixtures where the weights follow a Normalized Random Measure with Independent Increments and derive two samplers. The Slice 1 sampler writes the likelihood contribution $\prod_{j=1}^{t} w_{s_j} k(y_j \mid \theta_{s_j})$ in the following way

$$\prod_{j=1}^{t} \mathrm{I}(u_j < J_{s_j}) k(y_j \mid \theta_{s_j}) \exp\left\{-v_j \sum_{k=1}^{\infty} J_k\right\} \tag{4}$$

where $\mathrm{I}(\cdot)$ is the indicator function. Integrating out $v_1, \ldots, v_t$ and $u_1, \ldots, u_t$ leads to the correct form. The Slice 2 sampler writes the likelihood contribution in the alternative from

$$\prod_{j=1}^{t} \frac{\mathrm{I}(u < \alpha_t)}{\alpha_t} J_{s_j} k(y_j \mid \theta_{s_j}) \exp\left\{-v_j \sum_{k=1}^{\infty} J_k\right\}.$$

where $\alpha_t = \min\{\hat{J}_j \mid j = 1, \ldots, K_t\}$ and, using the notation of section 3, $\hat{J}_1, \ldots, \hat{J}_{K_t}$ are the sizes of jumps which have been allocated observations. The introduction of the latent variables $u_1, \ldots, u_t$ in Slice 1 and u in Slice 2 leads to a full conditional distribution for s_t which is discrete with a finite number of possible values.

The forms of the likelihood introduced in Slice 1 and Slice 2 are also convenient for sequential Monte Carlo methods since the number of latent parameters grows with the number of observations. However, it is not immediately clear how to sample from the joint distribution of v_t and u_t in Slice 1 or v_t in Slice 2 conditional on the values generated for each particle at previous steps of the algorithm. The following method is a simple solution which works for both Slice 1 and Slice 2. In Slice 1, we firstly integrate all jumps ($\hat{J}$ and $\tilde{J}$ defined at the end of section 3) from the model then the latent variable v_t is sampled using the method for a conjugate model. The latent variable u_t is sampled by first simulating another latent variable d_t according to the conditional distribution of s_t given in (3). If d_t is associated with a new jump then a new value is drawn from the centring distribution H and added to $\hat{\theta}$. The points in $\hat{J}$ are then simulated conditional on $s_{1:(t-1)}$ and d_t and associated with $\hat{\theta}$, and finally simulating u_t from $\mathrm{U}\left(0, \hat{J}_{d_t}\right)$. This allows us to simulate the R_t jumps with size in (u_t, ∞) and no observation allocated. These are denoted $\tilde{J}_1, \ldots, \tilde{J}_{R_t}$ which follow a Poisson process with intensity $\exp\left\{-J \sum_{j=1}^{t} v_j\right\} \eta(J)$. Values of $\tilde{\theta}$ are simulated from H and associated with each point of $\tilde{J}$. The sample of u_t, $\hat{J}$ and $\tilde{J}$ are from

the joint distribution of u_t and J (restricted to (u_t, ∞)) conditional on previous values. This allows us to sample s_t from its conditional distribution defined by (4). Once all particles have been sampled, they are re-weighted. Algorithm 3 describes all necessary steps. The algorithm for Slice 1 can be easily adapted to the latent variables construction in Slice 2. Firstly, the sampling step for u_t in Slice 1 can be replaced by the following sampling step for u, simulated $u \sim \mathrm{U}(0, \beta_t)$ where β_t is the minimum of $J_{s_1}, \ldots, J_{s_{t-1}}$ and J_{d_t} and $\tilde{J}$ is now from a Poisson process with intensity $\exp\{-J \sum_{j=1}^{t} v_j\}\eta(J)$ restricted to the interval (u, ∞). The allocation s_t is then simulated from the conditional distribution $q(s_t = k) \propto \max\{J_k, \alpha_{t-1}\} k(y_t \mid \theta_k)$. Once all particles have been sampled, they are re-weighted. Algorithm 4 describes the full method.

6 Extensions to the Sequential Monte Carlo methods

6.1 Resampling

Both sequential Monte Carlo methods for non-conjugate NRMI models can suffer from low effective sample sizes. This problem is caused by several factors. Firstly, the values of $s_1^{(i)}, \ldots, s_t^{(i)}$ and $\theta_1^{(i)}, \ldots, \theta_{K_t^{(i)}}^{(i)}$ (for the non-conjugate model) are fixed after the t-th iteration leading to a lack of heterogeneity in values of some $\theta_k^{(1)}, \ldots, \theta_k^{(N)}$ (the problem of particle depletion). Secondly, new values of $\theta_k^{(i)}$ are effectively proposed from their prior H and this can lead to drawn values which are not consistent with y_t. The first problem has been widely considered in the literature (Gilks and Berzuini, 2001; Chopin, 2002) and can be addressed by re-sampling previously sampled values of $\hat{\theta}_k^{(i)}$ from their full conditional distribution at step 1c) in Algorithms 3 and 4. The density is proportional to

$$h\left(\hat{\theta}_k^{(i)}\right) k\left(y_t|\hat{\theta}_k^{(i)}\right)^{\mathrm{I}\left(d_t^{(i)}=k\right)} \prod_{j=1}^{t-1} k\left(y_j|\hat{\theta}_k^{(i)}\right)^{\mathrm{I}\left(s_j^{(i)}=k\right)}.$$

The second problem is also standard to many particle filters and can be addressed by proposing the values of $\tilde{\theta}$ in step 1f) of Algorithms 3 and 4 from a density that depend on the current observation. The Auxiliary Particle Filter (Pitt and Shephard, 1999) would choose $h^{adp}\left(\tilde{\theta}_k^{(i)}\right) \propto h\left(\tilde{\theta}_k^{(i)}\right) k\left(y_t|\tilde{\theta}_k^{(i)}\right)$. If this choice cannot be sampled straightforwardly then a choice of $h^{adp}\left(\tilde{\theta}_k^{(i)}\right)$ that approximates this distribution could be used. The

For $t = 1, \ldots, n$, perform steps (1) and (2)

1. For $i = 1, \ldots, N$, perform steps (a)–(g)

   (a) Sample $v_t^{(i)}$ from the distribution $v_t^{(i)} \mid s_{1:(t-1)}^{(i)}, v_{1:(t-1)}^{(i)}$

   (b) Sample d_t from the distribution proportional to $p\left(s_t^{(i)} \mid s_{1:(t-1)}^{(i)}, v_{1:t}^{(i)}\right)$, If $d_t = K_{t-1}^{(i)} + 1$, simulate $\tilde{\theta}_{K_{t-1}^{(i)}+1}^{(i)} \sim H$.

   (c) Sample $\hat{J}_1^{(i)}, \ldots, \hat{J}_{K_t^{(i)}}^{(i)}$ (if $d_t \leq K_{t-1}^{(i)}$) or $\hat{J}_1^{(i)}, \ldots, \hat{J}_{K_t^{(i)}+1}^{(i)}$ (if $d_t = K_{t-1}^{(i)} + 1$). The density of $\hat{J}_k^{(i)}$ is proportional to

   $$\left(\hat{J}_k^{(i)}\right)^{m_{k,t-1}^{(i)} + \mathrm{I}(d_t = k)} \exp\left\{-\hat{J}_k^{(i)} \sum_{j=1}^{t} v_j\right\} \eta\left(\hat{J}_k^{(i)}\right).$$

   (d) Sample $u_t^{(i)} \sim \mathrm{U}\left(0, \hat{J}_{d_t}^{(i)}\right)$

   (e) Sample $\tilde{J}_1^{(i)}, \ldots, \tilde{J}_{R_t^{(i)}}^{(i)}$ from a Poisson process on $\left(u_t^{(i)}, \infty\right)$ with intensity $\exp\left\{-J \sum_{j=1}^{t} v_j^{(i)}\right\} \eta(J)$. Simulate $\tilde{\theta}_1^{(i)}, \ldots, \tilde{\theta}_{R_t^{(i)}}^{(i)} \overset{i.i.d.}{\sim} H$.

   (f) Let $J^{(i)} = \left\{\hat{J}^{(i)}, \tilde{J}^{(i)}\right\}$ and $\theta^{(i)} = \left\{\hat{\theta}^{(i)}, \tilde{\theta}^{(i)}\right\}$. Sample $s_t^{(i)}$ according to

   $$p(s_t^{(i)} = k) \propto \mathrm{I}\left(J_k > u_t^{(i)}\right) k\left(y_t \mid \theta_k^{(i)}\right), \qquad (k = 1, \ldots, K_t^{(i)} + R_t^{(i)}).$$

   (g) Calculate the unnormalized weight

   $$\psi_t^{(i)} = \frac{\sum_{k=1}^{K_t^{(i)} + R_t^{(i)}} \mathrm{I}\left(J_k^{(i)} > u_t^{(i)}\right) k\left(y_t \mid \theta_k^{(i)}\right)}{\sum_{k=1}^{K^{(i)} + R_t^{(i)}} \mathrm{I}\left(J_k^{(i)} > u_t^{(i)}\right)}.$$

2. Re-weight the particles according to the weights $\zeta_t^{(i)} = \frac{\psi_t^{(i)}}{\sum_{i=1}^{N} \psi_t^{(i)}} \quad (i = 1, \ldots, N)$.

Algorithm 3: Slice 1 SMC algorithm for non-conjugate NRMI mixture models

For $t = 1, \ldots, n$, perform steps (1) and (2)

1. For $i = 1, \ldots, N$, perform steps (a)–(g)

   (a) Sample $v_t^{(i)}$ from the distribution $v_t^{(i)} \mid s_{1:(t-1)}^{(i)}, v_{1:(t-1)}^{(i)}$

   (b) Sample d_t from the distribution proportional to $p\left(s_t^{(i)} \mid s_{1:(t-1)}^{(i)}, v_{1:t}^{(i)}\right)$. If $d_t = K_{t-1}^{(i)} + 1$, simulate $\tilde{\theta}_{K_{t-1}^{(i)}+1}^{(i)} \sim H$.

   (c) Sample $\hat{J}_1^{(i)}, \ldots, \hat{J}_{K_t^{(i)}}^{(i)}$ (if $d_t \leq K_{t-1}^{(i)}$) or $\hat{J}_1^{(i)}, \ldots, \hat{J}_{K_t^{(i)}+1}^{(i)}$ (if $d_t = K_{t-1}^{(i)} + 1$). The density of $\hat{J}_k^{(i)}$ is proportional to

   $$\left(\hat{J}_k^{(i)}\right)^{m_{k,t-1}^{(i)} + I(d_t = k)} \exp\left\{-\hat{J}_k^{(i)} \sum_{j=1}^{t} v_j^{(i)}\right\} \eta\left(\hat{J}^{(i)}\right).$$

   (d) Let $\alpha_{t-1}^{(i)} = \min\left\{\hat{J}_1^{(i)}, \ldots, \hat{J}_{K_{t-1}^{(i)}}^{(i)}\right\}$ and $\beta_t^{(i)} = \min\left\{\hat{J}_1^{(i)}, \ldots, \hat{J}_{K_{t-1}^{(i)}}^{(i)}, \hat{J}_{d_t^{(i)}}^{(i)}\right\}$.

   (e) Sample $u^{(i)} \sim \mathrm{U}\left(0, \beta_t^{(i)}\right)$

   (f) Sample $\tilde{J}_1^{(i)}, \ldots, \tilde{J}_{R_t^{(i)}}^{(i)}$ from a Poisson process on $\left(u^{(i)}, \infty\right)$ with intensity $\exp\left\{-J \sum_{j=1}^{t} v_j^{(i)}\right\} \eta(J)$. Simulate $\tilde{\theta}_1^{(i)}, \ldots, \tilde{\theta}_{R_t^{(i)}}^{(i)} \overset{i.i.d.}{\sim} H$.

   (g) Let $J^{(i)} = \left\{\hat{J}^{(i)}, \tilde{J}^{(i)}\right\}$ and $\theta^{(i)} = \left\{\hat{\theta}^{(i)}, \tilde{\theta}^{(i)}\right\}$. Sample $s_t^{(i)}$ according to

   $$q\left(s_t^{(i)} = k\right) \propto \max\left\{J_k^{(i)}, \alpha_{t-1}^{(i)}\right\} k\left(y_t \mid \theta_k^{(i)}\right).$$

   (h) Calculate the unnormalized weight

   $$\psi_t^{(i)} = \frac{\sum_{k=1}^{K_t^{(i)} + R_t^{(i)}} \max\left\{J_k^{(i)}, \alpha_{t-1}^{(i)}\right\} k\left(y_t \mid \theta_k^{(i)}\right)}{\sum_{k=1}^{K^{(i)}} \max\left\{J_k^{(i)}, \alpha_{t-1}^{(i)}\right\}}$$

2. Re-weight the particles according to the weights $\zeta_i = \frac{\psi_t^{(i)}}{\sum_{i=1}^{N} \psi_t^{(i)}} \quad (i = 1, \ldots, N)$.

Algorithm 4: Slice 2 SMC algorithm for non-conjugate NRMI mixture models

value of $\tilde{\zeta}_t^{(i)}$ is adjusted to

$$\psi_t^{(i)} = \frac{\sum_{k=1}^{K_t^{(i)}+R_t^{(i)}} \mathrm{I}\left(J_k^{(i)} > u_t^{(i)}\right) k\left(y_t \mid \theta_k^{(i)}\right) h\left(\tilde{\theta}_k^{(i)}\right) / h^{adp}\left(\tilde{\theta}_k^{(i)}\right)}{\sum_{k=1}^{K_t^{(i)}+R_t^{(i)}} \mathrm{I}\left(J_k^{(i)} > u_t^{(i)}\right)}.$$

in Algorithm 3 and to

$$\psi_t^{(i)} = \frac{\sum_{k=1}^{K_t^{(i)}+R_t^{(i)}} \max\left\{J_k^{(i)}, \alpha_{t-1}^{(i)}\right\} k\left(y_t \mid \theta_k^{(i)}\right) h\left(\tilde{\theta}_k^{(i)}\right) / h^{adp}\left(\tilde{\theta}_k^{(i)}\right)}{\sum_{k=1}^{K_t^{(i)}+R_t^{(i)}} \max\left\{J_k^{(i)}, \alpha_{t-1}^{(i)}\right\}}$$

in Algorithm 4

All sequential Monte Carlo algorithms for mixture models involve sampling $s_t^{(i)}$ at the t-th iteration from a finite, discrete distribution. The number of elements in this discrete distribution is the number of distinct values which is random. In the Dirichlet process with mass parameter M, $E[K_t] = M \log t$ for large t. Therefore, the number of operations needed to sample $s_1, \ldots, s_t$ is approximately $M \sum_{j=1}^{t} \log j = M \log(t!)$ which is, using Stirling's formula, approximately $M\left(t \log t - t + \log(2\pi t)/2\right)$ so any SMC algorithm for Dirichlet process mixture models has computational complexity $O(n \log n)$. Similarly, the number of distinct values will grow with any NRMI and so the computational complexity will be greater than the usual $O(n)$ of a particle filter. Performing updates of $s_{1:(t-1)}$ for all t would lead to an algorithm with computational complexity $O(n^2)$. However, if update only occur at pre-selected values of t which scale in a suitable way then the computational complexity for the algorithm can be unchanged. For example, updating at $\lceil k^i \rceil$, $(i = 1, 2, \ldots)$ for $k > 1$ leads to an algorithm that scales like $O(n \log n)$ for the Dirichlet process mixture model.

6.2 Parameter estimation

In many applications of Bayesian nonparametric methods, there are static parameters which we would like to infer. For example, the parameter ϕ in (1) is a static parameter. Similarly, there may be parameters that control the random probability measure (such as the mass parameter M in the Dirichlet process) or the centring distribution H may have parameters. The estimation of static parameters in sequential Monte Carlo samplers is difficult. The simplest method include the parameters as extra dimensions of the particle. However, this can lead to particle degeneracy and poor estimation of the posterior distribution of

the parameters. Alternative, the parameters could be integrated out from the model. An alternative method adopted in this paper is to update the parameters using Gibbs step, either sampling directly from the full conditional or using a Metropolis-Hastings scheme such as random walk. Updating all parameters at every step may be computational intensive so we suggest, firstly, using sufficient statistics where possible and, secondly, if this isn't possible then parameters are updated for values of j equal to $x^k \quad (k = 1, 2, \dots)$.

6.3 Marginal likelihood estimation

Marginal likelihood of models plays a crucial role in the calculation of Bayes factor for hypothesis testing or the combination of a small number of models using Bayesian model averaging. The estimation of marginal likelihood for nonparametric models has been particularly challenging. Basu and Chib (2003) describe a method for approximating marginal likelihood for MCMC output but this can be time-consuming. Del Moral (2004) shows that sequential Monte Carlo methods lead to a simple, unbiased estimate of the marginal likelihood which is

$$\sum_{t=1}^{n} \left(\frac{1}{N} \sum_{i=1}^{N} \psi_t^{(i)} \right).$$

This allows the comparison of the same model with different nonparametric priors or different models which each have a nonparametric component.

7 Illustrations

7.1 Comparison of SMC samplers

The mixture of normals model is one of the most popular in Bayesian nonparametrics and is a natural testing ground for the methods developed in this paper. We use the prior of Griffin (2010) who writes the model for observations $y_1, \dots, y_n$ as

$$\begin{aligned} y_t &\sim \text{N}(\mu_t, a\sigma^2), \qquad t = 1, \dots, n \\ \mu_t &\sim \text{NGG}(\gamma, 1, M, H), \qquad t = 1, \dots, n \end{aligned} \tag{5}$$

where H is a normal distribution with mean μ_0 and variance $(1-a)\sigma^2$. The methods are applied to two datasets: the ever-popular galaxy data and the log acidity data. The data are standardized to have mean 0 and variance 1 and we set $\mu_0 = 0$ and $\sigma = 1$ in the model.

The parameter a is fixed to 0.03 for the galaxy data and 0.16 for the log acidity (these are similar to the values estimated by Griffin, 2010). The data were randomly permuted and the sequential Monte Carlo algorithms run with 5 000 particles. The effective sample size is calculated using the method of Carpenter et al. (1999).

An initial comparison was made using the algorithm for conjugate Dirichlet process (which is the standard sequential Monte Carlo algorithm for conjugate Dirichlet processes), Slice 1 and Slice 2 for non-conjugate Dirichlet process models. The latter two algoritms were also combined with two extensions described in section 6. Firstly, updating of the parameters (U) and an Auxiliary Particle Filter proposal (A), which can be simply sampled since the model is conjugate. The number of clusters was used as the parameter of interest

Table 1: The effective sample size of estimating the posterior mean number of clusters from 5000 particles with a Dirichlet process prior.

Algorithm	Galaxy	Log Acidity
Conjugate	959	710
Slice 1	4	74
Slice 2	12	137
Slice 1 + U	16	98
Slice 2 + U	62	480
Slice 1 + A + U	78	131
Slice 2 + A + U	233	398

for the effective sample size calculations. The results are given in Table 1. The conjugate sampler performs well for both data sets but the non-conjugate sampler have a wide range of effective sample sizes with Slice 2 always outperforming Slice 1. In all cases, updating the parameters in the Dirichlet process increases the effective sample size (by a factor of 4 for the two datasets with Slice 2). The introduction of a proposal which is adapted to the current data point in the algorithm also leads to improvement in the effective sample sizes. Slice 1 will be removed from subsequent comparisons since it is always outperformed by Slice 2.

Section 6 also describes a method for adding updating steps for s without changing the computational complexity of the algorithm. Results are presented in Table 2 with the factor k chosen to be 1.5 and 2 ($k = 1.5$ has more update steps) and show that the introduction of the step leads to larger effective sample sizes. In the galaxy data, the choice of k leads to little difference. In contrast, for the log acidity data, the extra steps introduced for $k = 1.5$,

Table 2: The effective sample size of estimating the posterior mean number of clusters from 5000 particles with a Dirichlet process prior with updating of s

	Galaxy		Log Acidity	
Algorithm	$k = 1.5$	$k = 2$	$k = 1.5$	$k = 2$
Conjugate	1630	1704	1392	1062
Slice 2	35	31	515	268
Slice 2 + U	128	136	683	532
Slice 2 + A + U	424	414	760	475

compared $k = 2$, leads to a clearly larger effective sample size.

Table 3: The effective sample size of estimating the posterior mean number of clusters from 5000 particles with a Normalized Generalized Gamma process prior with updating of s.

	Galaxy		Log Acidity	
Algorithm	$k = 1.5$	$k = 2$	$k = 1.5$	$k = 2$
Conjugate	1630	1704	1243	1186
Slice 2 + U	118	136	670	494
Slice 2 + A + U	424	415	959	800

Results for the model with a Normalized Generalized Gamma prior for the mixing distribution are given in Table 3 with updating of s in the algorithm. The estimates of the effective sample sizes are roughly similar to the Dirichlet process mixture model with the algorithm for the conjugate model showing large effective sample sizes but also the algorithms for the non-conjugate models, particularly Slice 2 with adaptation and updating, showing good performance.

Table 4: The effective sample size of estimating the posterior mean number of a from 5000 particles with a Dirichlet process prior.

	Galaxy		Log Acidity	
Algorithm	$k = 1.5$	$k = 2$	$k = 1.5$	$k = 2$
Conjugate	606	1100	632	1554
Slice 2 + U	82	110	343	376
Slice 2 + A + U	222	343	574	447

The previous results assumed a fixed value for the parameter a which plays a crucial role in determining the modality and shape of the distribution. Often, we would want to

estimate this parameter. Table 4 shows results for the Dirichlet process mixture model with a given a uniform prior on $(0, 1)$. The effective sample size when the parameter of interest is the posterior mean of a. The parameter a can be updated using sufficient statistics and so was updated at every iteration. The results indicate that the algorithm produces good effective sample sizes in all cases. The results for a Normalized Generalized Gamma prior

Table 5: The effective sample size of estimating the posterior mean number of a from 5000 particles with a Normalized Generalized Gamma process prior.

	Galaxy		Log Acidity	
Algorithm	$k = 1.5$	$k = 2$	$k = 1.5$	$k = 2$
Conjugate	2075	1354	1399	757
Slice 2 + U	226	239	286	451
Slice 2 + A + U	329	519	425	503

on the mixing distribution are shown in Table 5 indicating broadly similar pattern of results to the Dirichlet process case with slightly larger effective sample size values. These results indicate that these sequential Monte Carlo algorithms gives good performance for posterior computation.

7.2 Testing a parametric model against a nonparametric alternative

The problem of testing a parametric model against a nonparametric alternative using Bayesian methods has received some attention in the literature. Carota and Parmigiani (1996) use a Dirichlet process based (rather than mixture of Dirichlet processes based) method whereas Berger and Guglielmi (2001) uses a method based on Polya trees. Consistency issue are considered by Dass and Lee (2004). More recently, McVinish et al. (2009) have proposed a method using mixtures of triangular distributions and considered its consistency. A different approach to testing a normal distribution uses a mixture of normal distributions to specify the nonparametric alternative distribution using the model in (5). The problem is slighly simplified by subtracting the sample mean from the data before analysis and assuming that μ_0 (the overall mean) is 0. The variance parameter σ^2 is given the standard non-informative prior, $p(\sigma^{-2}) \propto \sigma^{-2}$, in both the parametric and nonparametric models. Therefore, the nonparametric model is centred over the parameteric model. The models can be compared using Bayes factor. Let $p(y \mid H_0)$ be the marginal likelihood under the normal model and $p(y \mid H_1)$ be the marginal likelihood under the nonparametric model

then the Bayes factor in favour of the parametric model is

$$\frac{p(y \mid H_0)}{p(y \mid H_1)}.$$

The marginal likelihood under the parametric model can be calculated analytically and the marginal likelihood under the nonparametric model is calculated using the method in section 6.3. The the conjugate sampler with updating of s and $k = 2$ was used. Two examples were considered: data simulated from a standard normal distribution and the galaxy data. Clearly, the Bayes factor should favour the parametric model for the simulated data and the nonparametric model for the galaxy data (since the data have a multi-modal distribution). The Bayes factor in favour of the parametric model was estimated to be 1.3 for the standard normal data whereas the Bayes factor in favour of the nonparametric model was estimated to be -32.3 for the galaxy data.

7.3 Clustering of time series

Methods for Bayesian clustering of economic time series have been proposed by several authors including Frühwirth-Schnatter and Kaufmann (2008) and Bauwens and Rombouts (2007). An alternative approach is a simple non-conjugate model which assumes that

$$\begin{aligned} y_{i,t} &= \mu_i + \rho_i(y_{i,t-1} - \mu_i) + \epsilon_{i,t}, \qquad (i = 1, \ldots, n; t = 1, \ldots, T) \\ \epsilon_{i,t} &\overset{i.i.d.}{\sim} \text{N}(0, \sigma_i^2), \qquad (i = 1, \ldots, n; t = 1, \ldots, T) \\ (\mu_i, \sigma_i^2, \rho_i) &\sim F, \qquad (i = 1, \ldots, n) \end{aligned}$$

where F is given a Normalized Generalized Gamma process with parameters $M = 2$, $a = 0.2$ and $\gamma = 1$. The process is centred over the following distribution $\mu \sim \text{N}(\mu_0, \sigma_0^2)$, $\sigma^2 \sim \text{IG}(a_{\sigma^2}, b_{\sigma^2})$ and $\rho \sim \text{U}(-1, 1)$. The model assumes that the time series follow independent AR(1) processes with the time series clustered according to their parameters.

The model is applied to annual per capita GDP growth rates for 62 NUTS2 European regions from 1995 to 2004. The chosen hyperparameter values were $\mu_0 = 0$, $\sigma_0^2 = 0.01$, $a_{\sigma^2} = 3$ and $b_{\sigma^2} = 0.02$ (implying a prior mean of 0.01). The Slice 2 algorithm was run with 20 000 particles and $k = 1.5$. Figure 1 shows the posterior adjacency matrix whose (i, j)-th entry is the posterior probability that $s_i = s_j$ (the data has been re-arranged to more clearly show the structure of the clustering). The graph indicates that there are two clearly defined clusters in the bottom left-hand corner (Group 1) and upper right-hand corner (Group 2). There are also some points which do not completely fit into either cluster (in the middle)

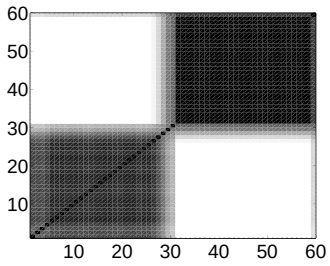

Figure 1: The posterior adjacency matrix for the NUTS2 data.

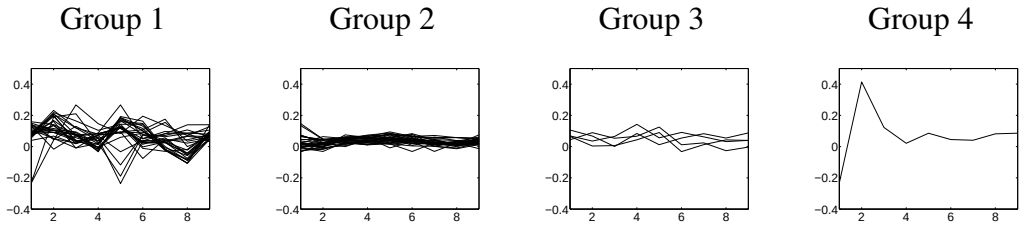

Figure 2: The time plots of the data of the four clusters found in the NUTS2 data.

(Group 3) and one observations which does not fit into either cluster (Group 4). Plots of the time series in these groups are shown in Figure 2. The consistency of growth across the regions grows from Group 2 to Group 3 to Group 1. Group 4 identifies an unusual the time series.

8 Discussion

The use of sequential Monte Carlo methods for the fitting of infinite mixture models is attractive since these can potentially avoid the problems of Gibb sampling from a potentially multi-modal posterior. The algorithms developed in this paper represent a viable alternative to Markov chain Monte Carlo methods when the mixing distribution is given a Normalized Random Measure with Independent Increments prior. The combination of these methods with Particle Markov chain Monte Carlo methods (Doucet et al., 2010) is a potentially powerful method for fitting highly structured infinite mixture model where the weights are constructed by normalization and allowed to depend on time or covariates.

References

Basu, S. and S. Chib (2003). Marginal likelihood and Bayes factors for Dirichlet process mixture models. *Journal of the American Statistical Association 98*, 224–235.

Bauwens, L. and J. V. K. Rombouts (2007). Bayesian clustering of many GARCH models. *Econometric Reviews 26*, 365–386.

Berger, J. and A. Guglielmi (2001). Testing of a parametric model versus nonparametric alternatives. *Journal of the American Statistical Association 96*, 174–184.

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

Brix, A. (1999). Generalized gamma measures and shot-noise Cox processes. *Adv. in Appl. Probab. 31*, 929–953.

Carota, C. and G. Parmigiani (1996). On Bayes Factors for Nonparametric Alternatives. In J. M. Bernardo, J. O. Berger, A. P. Dawid, and A. F. M. Smith (Eds.), *Bayesian Statistics 5*, London, pp. 508–511. Oxford University Press.

Carpenter, J., P. Clifford, and P. Fearnhead (1999). An improved particle filter for non-linear problems. *IEE Proceedings - Radar, Sonar and Navigation 146*, 2–7.

Carvalho, C. M., H. F. Lopes, N. G. Polson, and M. A. Taddy (2010). Particle Learning for General Mixtures. *Bayesian Analysis 5*, 709–740.

Chopin, N. (2002). A sequential particle filter for static models. *Biometrika 89*, 539–551.

Dass, S. C. and J. Lee (2004). A note on the consistency of Bayes factors for testing point null versus non-parametric alternatives. *Journal of Statistical Planning and Inference 119*, 143–152.

Del Moral, P. (2004). *Feynman-Kac Formulae: Genealogical and Interacting Particle Systems with Applications*. Springer.

Doucet, A., C. Andrieu, and R. Holenstein (2010). Particle Markov chain Monte Carlo methods (with discussion). *Journal of the Royal Statistical Society Series B 72*, 269–342.

Escobar, M. D. and M. West (1995). Bayesian density estimation and inference using mixtures. *J. Amer. Statist. Assoc. 90*, 577–588.

Fearnhead, P. (2004). Particle filters for mixture models with an unknown number of components. *Statistics and Computing 14*, 11–21.

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

Frühwirth-Schnatter, S. and S. Kaufmann (2008). Model-based clustering of multiple time series. *Journal of Business and Economic Statistics 26*, 78–89.

Gilks, W. R. and C. Berzuini (2001). Following a moving target: Monte Carlo inference for dynamic Bayesian models. *Journal of the Royal Statistical Society B 63*, 127–46.

Griffin, J. E. (2010). Default priors for density estimation with mixture models. *Bayesian Analysis 5*(1), 45–64.

Griffin, J. E. (2011). The Ornstein-Uhlenbeck Dirichlet process and other time-varying processes for Bayesian nonparametric inference. *Journal of Statistical Planning and Inference*, forthcoming.

Griffin, J. E. and S. G. Walker (2011). Posterior simulation of Normalised Random Measure mixtures. *Journal of Computational and Graphical Statistics 20*, 241–259.

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

James, L., A. Lijoi, and I. Prünster (2009). Posterior Analysis for Normalized Random Measures with Independent Increments. *Scandinavian Journal of Statistics 36*, 76–97.

Kalli, M., J. E. Griffin, and S. G. Walker (2011). Slice sampling mixture models. *Statistics and Computing 21*, 93–105.

Lijoi, A., R. H. Mena, and I. Prünster (2005). Hierarchical mixture modeling with normalized inverse-Gaussian priors. *J. Amer. Statist. Assoc. 100*, 1278–1291.

Lijoi, A., R. H. Mena, and I. Prünster (2007). Controlling the reinforcement in Bayesian non-parametric mixture models. *J. R. Stat. Soc. Ser. B 69*, 715–740.

Liu, J. S. (1996). Nonparametric hierarchical Bayes via sequential imputation. *Annals of Statistics 24*, 910–930.

MacEachern, S. N., M. A. Clyde, and J. Liu (1999). Sequential importance sampling for nonparametric Bayes models: The next generation. *Canadian Journal of Statistics 27*, 251–267.

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

McVinish, R., J. Rousseau, and K. Mengersen (2009). Bayesian Goodness of Fit Testing with Mixtures of Triangular Distributions. *Scandivavian Journal of Statistics 36*, 337–354.

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

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

Pitt, M. K. and N. Shephard (1999). Filtering via Simulation: Auxiliary Particle Filters. *Journal of the American Statistical Association 94*, 590–599.

Rao, V. and Y. W. Teh (2009). Spatial Normalized Gamma Processes. In *Advances in Neural Information Processing Systems*.

Ulker, Y., B. Gunsel, and A. Taylan Cemgil (2010). Sequential Monte Carlo Samplers for Dirichlet Process Mixtures. *Journal of Machine Learning Research 9*, 876–883.

Walker, S. G. (2007). Sampling the Dirichlet mixture model with slices. *Comm. Statist. Simulation Comput. 36*(1-3), 45–54.

A Appendix

A.1 Dirichlet process

The Dirichlet process (Ferguson, 1973) is a normalized Gamma process. Then

$$f(t,s,k) = M\Gamma(m_{k,t})\left(1+\sum_{j=1}^{s} v_j\right)^{-m_{k,t}}, \qquad L(v) = M\log\left(1+\sum_{i=1}^{t} v_i\right),$$

and

$$-\frac{d}{dv_n}\prod_{k=1}^{K_{t-1}} f(t-1,t,k)\exp\left\{-L_t\left(v\right)\right\} = (M+t-1)\prod_{k=1}^{K_{t-1}}\Gamma\left(m_{k,t-1}\right)\left(1+\sum_{j=1}^{t-1} v_j + v_t\right)^{-(M+t)}.$$

In the conjugate case, we have

$$p(s_{1:t}, v_{1:t}) = M^{K_t}\prod_{k=1}^{K_t}\Gamma\left(m_{k,t}\right)\left(1+\sum_{j=1}^{t} v_j\right)^{-(M+t)}$$

and

$$p(s_{1:(t-1)}, v_{1:t}) = M^{K_{t-1}}(M+t-1)\prod_{k=1}^{K_{t-1}}\Gamma\left(m_{k,t-1}\right)\left(1+\sum_{j=1}^{t} v_j\right)^{-(M+t)}$$

which leads to

$$p(s_t \mid s_{1:(t-1)}, v_{1:t}) = \frac{M^{K_t}\prod_{k=1}^{K_t}\Gamma\left(m_{k,t}\right)}{M^{K_{t-1}}(M+n-1)\prod_{k=1}^{K_{t-1}}\Gamma\left(m_{k,t-1}\right)}\left(1+\sum_{j=1}^{t} v_i\right).$$

The expression leads dirctly to the well-known Pólya urn scheme representation of the Dirichlet process (Blackwell and MacQueen, 1973),

$$p(s_t = k \mid s_{1:(t-1)}, v_{1:t}) = \begin{cases} \frac{m_j^{(t-1)}}{M+m-1} & \text{if } j \le K_{t-1} \\ \frac{M}{M+m-1} & \text{if } k = K_{t-1}+1 \end{cases}.$$

The conditional distribution of v_t has the form

$$p(v_t \mid s_{1:(t-1)}, v_{1:(t-1)}) = \frac{(M+t-1)}{\left(1+\sum_{j=1}^{t-1} v_i\right)^{-(M+t-1)}}\left(1+\sum_{j=1}^{t-1} v_i + v_t\right)^{-(M+t)}.$$

For $t = 1, \ldots, n$, perform steps (1) and (2)

1. For $i = 1, \ldots, N$, perform steps (a)–(g)

   (a) Sample $\tau_t^{(i)} \sim \text{Ga}\left(M + t - 1, \sum_{j=1}^{t-1} v_j^{(i)}\right)$ and $v_t^{(i)} \sim \text{Ex}\left(\tau_t^{(i)}\right)$.

   (b) Sample d_t according to the following probabilities

   $$p(d_t = k) \propto \begin{cases} m_{j,t-1}^{(i)} & \text{if } k \leq K_{t-1}^{(i)} \\ M & \text{if } k = K_{t-1}^{(i)} \end{cases}$$

   If $d_t = K_{t-1}^{(i)} + 1$, simulate $\tilde{\theta}_{K_{t-1}^{(i)}+1}^{(i)} \sim H$.

   (c) Sample $\hat{J}_1^{(i)}, \ldots, \hat{J}_{K_t^{(i)}}^{(i)}$ (if $d_t^{(i)} \leq K_{t-1}^{(i)}$) or $\hat{J}_1^{(i)}, \ldots, \hat{J}_{K_t^{(i)}+1}^{(i)}$ (if $d_t^{(i)} = K_{t-1}^{(i)} + 1$) according to $\hat{J}_k^{(i)} \sim \text{Ga}\left(m_{k,t-1}^{(i)} + \text{I}(d_t^{(i)} = k), 1 + \sum_{j=1}^{t} v_j\right)$.

   (d) Let $\alpha_{t-1}^{(i)} = \min\left\{\hat{J}_1^{(i)}, \ldots, \hat{J}_{K_{t-1}^{(i)}}^{(i)}\right\}$ and $\beta_t^{(i)} = \min\left\{\hat{J}_1^{(i)}, \ldots, \hat{J}_{K_{t-1}^{(i)}}^{(i)}, \hat{J}_{d_t^{(i)}}^{(i)}\right\}$.

   (e) Sample $u^{(i)} \sim \text{U}\left(0, \beta_t^{(i)}\right)$

   (f) Sample $\tilde{J}_1^{(i)}, \ldots, \tilde{J}_{R_t^{(i)}}^{(i)}$ from a Poisson process on $\left(u_t^{(i)}, \infty\right)$ with intensity $MJ^{-1} \exp\left\{-J\left(1 + \sum_{j=1}^{t} v_j^{(i)}\right)\right\}$. Simulate $\tilde{\theta}_1^{(i)}, \ldots, \tilde{\theta}_{R_t^{(i)}}^{(i)} \overset{i.i.d.}{\sim} H$.

   (g) Let $J^{(i)} = \left\{\hat{J}^{(i)}, \tilde{J}^{(i)}\right\}$ and $\theta^{(i)} = \left\{\hat{\theta}^{(i)}, \tilde{\theta}^{(i)}\right\}$. Sample $s_t^{(i)}$ according to

   $$q\left(s_t^{(i)} = k\right) \propto \max\left\{J_k^{(i)}, \alpha_{t-1}^{(i)}\right\} k\left(y_t \mid \theta_k^{(i)}\right).$$

   (h) Calculate the unnormalized weight

   $$\psi_t^{(i)} = \frac{\sum_{k=1}^{K_t^{(i)}+R_t^{(i)}} \max\left\{J_k^{(i)}, \alpha_{t-1}^{(i)}\right\} k\left(y_t \mid \theta_k^{(i)}\right)}{\sum_{k=1}^{K_t^{(i)}+R_t^{(i)}} \max\left\{J_k^{(i)}, \alpha_{t-1}^{(i)}\right\}}$$

2. Re-weight the particles according to the weights $\zeta_i = \frac{\psi_t^{(i)}}{\sum_{i=1}^{N} \psi_t^{(i)}} \quad (i = 1, \ldots, N)$.

Algorithm 5: Slice 2 SMC algorithm for non-conjugate DP mixture models

However, since $p(s_t = k \mid s_{1:(t-1)}, v_{1:t})$ does not depend on $v_{1:t}$ then it is not necessary to sample its value. It follows that the algorithm has exactly the same form as the standard Pólya urn scheme particle filter in Algorithm 1.

In non-conjugate Dirichlet process models, unlike the conjugate version, the value of v_t must be sampled which can done conveniently using the following scheme: simulate $\tau \sim \text{Ga}(M, 1 + \sum_{j=1}^{t-1} v_j)$ and then $v_t \sim \text{Ex}(\tau)$. The k-th jump in $\hat{J}$ at the t-th iteration is simulated from Ga $\left(m_{k,t-1}^{(i)} + \text{I}(d_t = k), 1 + \sum_{j=1}^{t} v_j^{(i)}\right)$. The points in $\tilde{J}$ are simulated from a Poisson process on $(\min\{u_t^{(i)}\}, \infty)$ with intensity $MJ^{-1}\exp\{-(1 + \sum_{j=1}^{t} v_j^{(i)})J$ which can be simulated using the method described in Griffin and Walker (2011).

A.2 Normalized Generalized Gamma process

The Normalized Generalized Gamma process has

$$f(t, s, k) = \frac{M}{\Gamma(1-\gamma)}\Gamma(m_{k,t}-\gamma)\left(\lambda + \sum_{j=1}^{s} v_j\right)^{-(m_{k,t}-\gamma)}, \qquad L(v) = \frac{M}{\gamma}\left(\left(\lambda + \sum_{j=1}^{t} v_j\right)^{\gamma} - \lambda^{\gamma}\right).$$

In the conjugate case, we have

$$p(s_{1:t}, v_{1:t}) = \left(\frac{M}{\Gamma(1-\gamma)}\right)^{K_t} \prod_{k=1}^{K_t} \Gamma\left(m_{k,t} - \gamma\right) \left(\lambda + \sum_{j=1}^{t} v_j\right)^{-(t-K_t\gamma)} \exp\left\{\frac{M}{\gamma}\left[\lambda^{\gamma} - \left(\lambda + \sum_{j=1}^{t} v_j\right)^{\gamma}\right]\right\}$$

$$\begin{aligned} p(s_{1:(t-1)}, v_{1:t}) = &\left(\frac{M}{\Gamma(1-\gamma)}\right)^{K_{t-1}} \exp\left\{\frac{M}{\gamma}\lambda^{\gamma}\right\} \prod_{k=1}^{K_{t-1}} \Gamma\left(m_{k,t-1} - \gamma\right) \left(\lambda + \sum_{j=1}^{t-1} v_j + v_t\right)^{K_{t-1}\gamma - t} \\ &\times \exp\left\{-\frac{M}{\gamma}\left(\lambda + \sum_{j=1}^{t-1} v_j + v_t\right)^{\gamma}\right\} \left\{M\left(\lambda + \sum_{j=1}^{t-1} v_j + v_t\right)^{\gamma} + (t-1) - K_{t-1}\gamma\right\} \end{aligned}$$

and so

$$\begin{aligned} p(s_t \mid s_{1:(t-1)}, v_{1:t}) = &\frac{\left(\frac{M}{\Gamma(1-\gamma)}\right)^{K_t}}{\left(\frac{M}{\Gamma(1-\gamma)}\right)^{K_{t-1}}} \frac{\prod_{k=1}^{K_t} \Gamma\left(m_{k,t} - \gamma\right) \left(\lambda + \sum_{j=1}^{t} v_j\right)^{K_t\gamma - t}}{\prod_{k=1}^{K_{t-1}} \Gamma\left(m_{k,t-1} - \gamma\right) \left(\lambda + \sum_{j=1}^{t} v_j\right)^{K_{t-1}\gamma - t}} \\ &\times \frac{1}{\left\{M\left(\lambda + \sum_{j=1}^{t} v_j\right)^{\gamma} + (t-1) - K_{t-1}\gamma\right\}} \end{aligned}$$

which can be expressed in terms of a Pólya urn scheme

For $t = 1, \ldots, n$, perform steps (1) and (2)

1. For $i = 1, \ldots, N$ perform steps (a)–(c)

   (a) Sample $v_t^{(i)}$ by inversion sampling from the distribution function

$$\frac{\left(\lambda + \sum_{j=1}^{t-1} v_j^{(i)} + v_t^{(i)}\right)^{-(t-1)+K_{t-1}^{(i)}\gamma} \exp\left\{-\frac{M}{\gamma}(\lambda + \sum_{j=1}^{t-1} v_j^{(i)} + v_t^{(i)})\right\}}{\left(\lambda + \sum_{j=1}^{t-1} v_j^{(i)}\right)^{-(t-1)+K_{t-1}^{(i)}\gamma} \exp\left\{-\frac{M}{\gamma}(\lambda + \sum_{j=1}^{t-1} v_j^{(i)})\right\}}.$$

   (b) Sample $s_t^{(i)}$ conditional on $y_{1:t}$, and $s_{1:(t-1)}^{(i)}$ from

$$q(k) \propto \begin{cases} (m_{k,t-1}^{(i)} - \gamma)\, p\left(y_t \mid y_{1:(t-1)}, s_{1:(t-1)}^{(i)}, s_t = k\right) & \text{if } k \leq K_{t-1}^{(i)} \\ M\left(\lambda + \sum_{j=1}^{t} v_j\right)^{\gamma} p\left(y_t \mid y_{1:(t-1)}, s_{1:(t-1)}^{(i)}, s_t = k\right) & \text{if } k = K_{t-1}^{(i)} + 1 \end{cases}.$$

   (c) Calculate the unnormalized weight

$$\begin{aligned} \psi_t^{(i)} =& M\left(\lambda + \sum_{j=1}^{t} v_j\right)^{\gamma} \text{pr}\left(y_t \mid s_{1:(t-1)}^{(i)}, s_t^{(i)} = K_{t-1}^{(i)} + 1, y_{1:(t-1)}\right) \\ &+ \sum_{k=1}^{K_t^{(i)}} (m_{k,t-1}^{(i)} - \gamma) \text{pr}\left(y_t \mid s_{1:(t-1)}^{(i)}, s_t^{(i)} = k, y_{1:(t-1)}\right). \end{aligned}$$

2. Re-weight the particles according to the weights $\zeta_i = \frac{\psi_t^{(i)}}{\sum_{i=1}^{N} \psi_t^{(i)}} \quad (i = 1, \ldots, N)$.

Algorithm 6: SMC algorithm for conjugate NGG process mixture

$$p(s_t = k) = \begin{cases} \frac{m_{k,t-1} - \gamma}{M\left(\lambda + \sum_{j=1}^{t} v_j\right)^{\gamma} + (t-1) - K_{t-1}\gamma} & \text{if } k \leq K_{t-1} + 1 \\ \frac{M\left(\lambda + \sum_{j=1}^{t} v_j\right)^{\gamma}}{M\left(\lambda + \sum_{j=1}^{t} v_j\right)^{\gamma} + (t-1) - K_{t-1}\gamma} & \text{if } k = K_{t-1} + 1 \end{cases}.$$

The distribution of $v_t \mid s_{1:(t-1)}, v_{1:(t-1)}$ has the distribution function

$$\frac{\left(\lambda + \sum_{j=1}^{t-1} v_j + v_t\right)^{-(t-1)+K_t\gamma} \exp\left\{-\frac{M}{\gamma}(\lambda + \sum_{j=1}^{t-1} v_j + v_t)\right\}}{\left(\lambda + \sum_{j=1}^{t-1} v_j\right)^{-(t-1)+K_t\gamma} \exp\left\{-\frac{M}{\gamma}(\lambda + \sum_{j=1}^{t-1} v_j)\right\}}$$

and so value of v_t can be simulated using inversion sampling. Algorithm 6 shows the full algorithm for the conjugate NGG mixture model.

In the non-conjugate NGG model, the k-th jump in $\hat{J}^{(i)}$ at the t-th iteration is simulated from Ga $\left(m_{k,t-1}^{(i)} + I(d_t = k) - \gamma, \lambda + \sum_{j=1}^{t} v_j^{(i)}\right)$. The points in $\tilde{J}^{(i)}$ are simulated from a Poisson process on $(\min\{u_t^{(i)}\}, \infty)$ with intensity $\frac{M}{\Gamma(1-\gamma)} J^{-1-\gamma} \exp\{-(\lambda + \sum_{j=1}^{t} v_j^{(i)})J$ which can be simulated using the method described in Griffin and Walker (2011).

For $t = 1, \ldots, n$, perform steps (1) and (2)

1. For $i = 1, \ldots, N$, perform steps (a)–(g)

   (a) Sample $v_t^{(i)}$ by inversion sampling from the distribution function

   $$\frac{\left(\lambda + \sum_{j=1}^{t-1} v_j^{(i)} + v_t^{(i)}\right)^{-(t-1)+K_{t-1}^{(i)}\gamma} \exp\left\{-\frac{M}{\gamma}(\lambda + \sum_{j=1}^{t-1} v_j^{(i)} + v_t^{(i)})\right\}}{\left(\lambda + \sum_{j=1}^{t-1} v_j^{(i)}\right)^{-(t-1)+K_{t-1}^{(i)}\gamma} \exp\left\{-\frac{M}{\gamma}(\lambda + \sum_{j=1}^{t-1} v_j^{(i)})\right\}}.$$

   (b) Sample d_t according to the following probabilities

   $$p(d_t = k) \propto \begin{cases} m_{j,t-1}^{(i)} - \gamma & \text{if } k \leq K_{t-1}^{(i)} \\ M\left(\lambda + \sum_{j=1}^{t} v_j^{(i)}\right)^{\gamma} & \text{if } k = K_{t-1}^{(i)} + 1 \end{cases}$$

   If $d_t = K_{t-1}^{(i)} + 1$, simulate $\hat{\theta}_{K_{t-1}^{(i)}+1}^{(i)} \sim H$.

   (c) Sample $\hat{J}_1^{(i)}, \ldots, \hat{J}_{K_t^{(i)}}^{(i)}$ (if $d_t \leq K_{t-1}^{(i)}$) or $\hat{J}_1^{(i)}, \ldots, \hat{J}_{K_t^{(i)}+1}^{(i)}$ (if $d_t = K_{t-1}^{(i)} + 1$) according to $\hat{J}_k^{(i)} \sim \text{Ga}\left(m_{k,t-1}^{(i)} + \text{I}(d_t = k) - \gamma, \lambda + \sum_{j=1}^{t} v_j^{(i)}\right)$.

   (d) Let $\alpha_{t-1}^{(i)} = \min\left\{\hat{J}_1^{(i)}, \ldots, \hat{J}_{K_{t-1}^{(i)}}^{(i)}\right\}$ and $\beta_t^{(i)} = \min\left\{\hat{J}_1^{(i)}, \ldots, \hat{J}_{K_{t-1}^{(i)}}^{(i)}, \hat{J}_{d_t^{(i)}}^{(i)}\right\}$.

   (e) Sample $u^{(i)} \sim \text{U}\left(0, \beta_t^{(i)}\right)$

   (f) Sample $\tilde{J}_1^{(i)}, \ldots, \tilde{J}_{R_t^{(i)}}^{(i)}$ from a Poisson process on $\left(u_t^{(i)}, \infty\right)$ with intensity $\frac{M}{\Gamma(1-\gamma)} J^{-1-\gamma} \exp\left\{-J\left(\lambda + \sum_{j=1}^{t} v_j^{(i)}\right)\right\}$. Simulate $\tilde{\theta}_1^{(i)}, \ldots, \tilde{\theta}_{R_t^{(i)}}^{(i)} \overset{i.i.d.}{\sim} H$.

   (g) Let $J^{(i)} = \left\{\hat{J}^{(i)}, \tilde{J}^{(i)}\right\}$ and $\theta^{(i)} = \left\{\hat{\theta}^{(i)}, \tilde{\theta}^{(i)}\right\}$. Sample $s_t^{(i)}$ according to

   $$q\left(s_t^{(i)} = k\right) \propto \max\left\{J_k^{(i)}, \alpha_{t-1}^{(i)}\right\} k\left(y_t \mid \theta_k^{(i)}\right)$$

   (h) Calculate the unnormalized weight

   $$\psi_t^{(i)} = \frac{\sum_{k=1}^{K_t^{(i)}+R_t^{(i)}} \max\left\{J_k^{(i)}, \alpha_{t-1}^{(i)}\right\} k\left(y_t \mid \theta_k^{(i)}\right)}{\sum_{k=1}^{K^{(i)}+R_t^{(i)}} \max\left\{J_k^{(i)}, \alpha_{t-1}^{(i)}\right\}}$$

2. Re-weight the particles according to the weights $\zeta_i = \frac{\psi_t^{(i)}}{\sum_{i=1}^{N} \psi_t^{(i)}} \quad (i = 1, \ldots, N)$.

Algorithm 7: Slice 2 SMC algorithm for non-conjugate NGG process mixture model