Sequential Monte Carlo methods for mixtures with normalized random measures with independent increments priors

J. E. Griffin[1]

Received: 12 November 2014 / Accepted: 1 November 2015 / Published online: 11 November 2015

Abstract Normalized random measures with independent increments are a general, tractable class of nonparametric prior. This paper describes sequential Monte Carlo methods for both conjugate and non-conjugate nonparametric mixture models with these priors. A simulation study is used to compare the efficiency of the different algorithms for density estimation and comparisons made with Markov chain Monte Carlo methods. The SMC methods are further illustrated by applications to dynamically fitting a nonparametric stochastic volatility model and to estimation of the marginal likelihood in a goodness-of-fit testing example.

Keywords Bayesian nonparametrics · Dirichlet process · Normalized generalized gamma process · Nonparametric stochastic volatility · Slice sampling · Particle Gibbs sampling

1 Introduction

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

$$\begin{aligned} y_t|\gamma_t &\sim k(y_t \mid \gamma_t, \phi) \qquad (t = 1, \dots, n) \qquad (1)\\ \gamma_t &\overset{i.i.d.}{\sim} G \qquad (t = 1, \dots, n) \end{aligned}$$

where $k(x \mid \gamma, \phi)$ is a probability density function for x with parameters γ and ϕ, and G is a distribution which is given a nonparametric prior. The most popular instance of this model is the Dirichlet process (DP) mixture model (Escobar and West 1995) where G is given a DP prior. This prior is computationally attractive but the choice can be restrictive and so tractable generalizations have been proposed. Ishwaran and James (2001) described the construction of stick-breaking priors (such as the Poisson–Dirichlet process) and James et al. (2009) discussed inference in the class of normalized random measures with independent increments (NRMI). In all these priors, G is a discrete distribution so that

$$G = \sum_{k=1}^{\infty} w_k \delta_{\theta_k} \qquad (2)$$

where δ_θ represents the Dirac measure that places measure 1 at θ, $w_k > 0 \quad (k = 1, 2, \dots)$ and $\sum_{k=1}^{\infty} w_k = 1$. I will write $\theta = (\theta_1, \theta_2, \theta_3, \dots)$ and $w = (w_1, w_2, w_3, \dots)$. Standard choices of prior often imply that $\theta_1, \theta_2, \dots \overset{i.i.d.}{\sim} H$ (whose density is h if H is continuous) and that θ and w are independent. The discreteness of G implies that the density of y_t is

$$p(y_t) = \sum_{k=1}^{\infty} w_k\, k(y_t \mid \theta_k, \phi) \quad (t = 1, \dots, n).$$

The construction of methods for posterior inference in nonparametric mixture models is challenging since an infinite number of parameters is involved and the posterior is typically analytically intractable. Many Markov chain Monte

Electronic supplementary material The online version of this article (doi:10.1007/s11222-015-9612-3) contains supplementary material, which is available to authorized users.

✉ J. E. Griffin
j.e.griffin-28@kent.ac.uk

[1] School of Mathematics, Statistics and Actuarial Science, University of Kent, Canterbury, Kent, UK

Carlo (MCMC) methods have been proposed using different representations of the nonparametric prior including: Pólya urn scheme representations (Escobar and West 1995; MacEachern and Müller 1998; Neal 2000; Favaro and Teh 2013), stick-breaking representations (Ishwaran and James 2001; Papaspiliopoulos and Roberts 2008; Walker 2007; Kalli et al. 2011) and normalized Lévy process representations (Griffin and Walker 2011). These allow effective inference for a wide-range of nonparametric prior for both conjugate model (where k and H are conjugate) and non-conjugate models in static inference problems.

The increasing range of applications of nonparametric models has lead to inferential problems and modelling situations which are not well-suited to MCMC methods. For example, in economics, Bayesian nonparametric mixture models have been applied to stochastic volatility modelling (Jensen and Maheu 2010; Delatola and Griffin 2011, 2013; Jensen and Maheu 2014) for a financial time series $y_1, \ldots, y_n$. These models assume a nonparametric mixture model for the unknown distribution $p(y_t|\sigma_t)$ where σ_t is a scale parameter which evolves according to a stochastic process. More generally, Caron et al. (2008) consider the use of DP mixtures in dynamic linear models. In these models, we may be interested in making inference about the unknown distribution at different time points. These results can be used either to perform dynamic inference or to compute model comparison measures such as log predictive scores (Geisser and Eddy 1979) or h-step ahead root mean squared error. The calculation of model marginal likelihoods, used in the calculation of Bayes factors, is another inferential problem that is difficult with MCMC methods and the estimation of marginal likelihoods for nonparametric models has been particularly challenging. Basu and Chib (2003) describe a method for approximating marginal likelihood from MCMC output but this can be time-consuming. Both these problems can be addressed using sequential Monte Carlo (SMC) methods.

In this paper, I will develop SMC methods for the wide-class of NRMI mixtures. SMC methods build an approximation of the posterior with observations $y_1, \ldots, y_t$ from an approximation of the posterior with observations $y_1, \ldots, y_{t-1}$. These have been heavily used with non-linear state space models in dynamic problems where the posterior distribution at each time point is needed for inference and prediction. Repeated application of this process leads to the posterior conditional on the full sample $y_1, \ldots, y_n$ and has been proposed as an alternative to MCMC methods for static problems (e.g.Chopin 2002). The model in (1) can be represented in terms of allocation variables $s_1, \ldots, s_n$ which link the observations to the components of the mixture model by $\gamma_t = \theta_{s_t}$. This alternative representation is

$$y_t|s_t = k \sim k(y_t \mid \theta_k, \phi) \quad (t = 1, \ldots, n) \tag{3}$$
$$p(s_t = k) = w_k \quad (t = 1, \ldots, n;\ k = 1, 2, \ldots).$$

This representation allows the nonparametric mixture model to be written in the form of a state space model where $k(y_t \mid \theta_{s_t}, \phi)$ is the observation equation, s_t is the state and w, θ and ϕ are static parameters. SMC methods for DP mixture models were 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 DP 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) extended their algorithm to a Sampling-Importance-Resampling algorithm (also known as a particle filter). Chopin (2002) described the application of a similar algorithm to finite mixture models. There has recently been renewed interest in SMC methods for nonparametric mixture models. Ulker et al. (2010) described elaborations of the algorithm of Fearnhead (2004) and Carvalho et al. (2010) described particle learning methods.

The paper is organised as follows. Section 2 reviews some previous work on SMC methods for DP mixture models and the wide class of NRMI's which generalise the DP. Section 3 discusses SMC methods for conjugate and non-conjugate NRMI mixtures. Section 4 briefly discusses the use of these algorithms in particle Markov chain Monte Carlo (PMCMC) samplers. Section 5 illustrates the use of these methods in a range of situations. Section 6 gives a brief discussion of the idea developed in the paper. The Online Appendix contains implementation details for two commonly used classes of NRMI's: the DP and the normalized generalized gamma process.

2 Background

In this section, I will review the use of SMC methods for DP mixture models and the wide class of NRMI's before considering the application of SMC methods to NRMI mixture models in Sect. 3. The notation $x_{i:j} = (x_i, \ldots, x_j)$ will be used as shorthand for vectors.

2.1 Sequential Monte Carlo methods for Dirichlet process mixture models

Fearnhead (2004) described an SMC algorithm for the model in (1) where G is a given a DP prior to define a DP mixture model. The Pólya urn scheme representation of the DP (Blackwell and MacQueen 1973) allows us to write the model in (3) as

$$y_t|s_t^\star = k \sim k(y_t \mid \theta_k^\star, \phi) \quad (t = 1, \ldots, n) \tag{4}$$
$$\mathrm{pr}\left(s_t^\star = k|s_{1:(t-1)}^\star\right) = \begin{cases} \frac{m_{t-1,k}}{M+t-1} & \text{if } 1 \leq k \leq K_{t-1} \\ \frac{M}{M+t-1} & \text{if } k = K_{t-1} + 1 \end{cases} \quad (t = 1, \ldots, n).$$

where $\theta^\star_{1:K_t}$ are the distinct values of $\gamma_{1:t}$, $m_{t,k} = \sum_{k=1}^{t} \mathrm{I}\left(\gamma_k = \theta^\star_k\right)$ and $s^\star_{1:t}$ are defined by $\gamma_t = \theta^\star_{s^\star_t}$. The allocation variables $s^\star_{1:n}$ are just a re-numbering of $s_{1:n}$ and $\theta^\star_{1:K_t}$ is a finite subset of θ.

If k and H are conjugate, we say that the DP mixture model is conjugate. In this case,

$$\mathrm{pr}\left(s^\star_t = k | s^\star_{1:(t-1)}, y_{1:t}\right) \propto \begin{cases} m_{t-1,k}\, k^\star_k(y_t \mid s^\star_{1:(t-1)}) & \text{if } 1 \le k \le K_{t-1} \\ M\, k^\star_{new}(y_t) & \text{if } k = K_{t-1}+1 \end{cases}$$

where

$$k^\star_k\left(y_t | s^\star_{1:(t-1)}\right) = \frac{\int k(y_t|\theta) \prod_{\{j|s^\star_j = k, 1\le j\le t-1\}} k(y_j \mid \theta) dH(\theta)}{\int \prod_{\{j|s^\star_j = k, 1\le j\le t-1\}} k(y_j \mid \theta) dH(\theta)}$$

and

$$k^\star_{new}(y_t) = \int k(y_t \mid \theta) dH(\theta).$$

The availability of this distribution allows an algorithm to be defined where N values $s^{(1)}_{1:t}, \ldots, s^{(N)}_{1:t}$ are sampled from $p(s_{1:t}|y_{1:t})$ sequentially in t. The value $s^{(i)}_{1:t}$ is called the value of $s_{1:t}$ in the ith particle and the notation $z^{(i)}$ is used generally to represent the value of z in the ith particle. The details are given in Algorithm 1. The algorithm can be very computationally efficient if $k^\star_k\left(y_t \mid s^\star_{1:(t-1)}\right)$ can be calculated using sufficient statistics (Fearnhead 2004).

```
For $t = 1, \ldots, n$, perform steps (1) and (2)
  1. For $i = 1, \ldots, N$ perform steps (a) and (b)
    (a) Sample $s^{\star\,(i)}_t$ conditional on $y_{1:t}$, and $s^{\star\,(i)}_{1:(t-1)}$ from
        $q(k) \propto \begin{cases} m^{(i)}_{k,t-1}\, k^\star_k\left(y_t \mid s^{\star\,(i)}_{1:(t-1)}\right) & \text{if } 1 \le k \le K^{(i)}_{t-1} \\ M\, k^\star_{new}(y_t) & \text{if } k = K^{(i)}_{t-1}+1 \end{cases}.$
    (b) Calculate the unnormalized weight
        $\xi^{(i)}_t = M k^\star_{new}(y_t) + \sum_{k=1}^{K^{(i)}_{t-1}} m^{(i)}_{k,t-1} k^\star_k\left(y_t \mid s^{\star\,(i)}_{1:(t-1)}\right).$
  2. Re-weight the particles according to the weights $\zeta_i = \frac{\xi^{(i)}_t}{\sum_{i=1}^N \xi^{(i)}_t}$ $(i = 1, \ldots, N)$.
```

Algorithm 1: SMC algorithm for conjugate DP mixture models

The algorithm can be extended to non-conjugate mixture models in several ways. Firstly, Algorithm 1 can be directly used if $k^\star_k\left(y_t \mid s^\star_{1:(t-1)}\right)$ and $k^\star_{new}(y_t)$ 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, values of $\theta^\star_{1:K_t}$ can be included directly in the algorithm (rather than integrating over their values) and a potential value of $\theta^{\star\,(i)}_{K^{(i)}_{t-1}+1}$ is generated from H (which is called θ_{new} here). This algorithm is summarized in Algorithm 2. The algorithm avoids the need to approximate some integrals but introduces static parameters into the SMC sampler with 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) in which $\theta^{\star\,(i)}_j$ for $j = 1, \ldots, K^{(i)}_t$ are updated for $i = 1, \ldots, N$ using an MCMC step such as a Metropolis-Hastings random walk step or a Gibbs step.

```
For $t = 1, \ldots, n$, perform steps (1) and (2)
  1. For $i = 1, \ldots, N$ perform steps (a) and (b)
    (a) Sample $\theta_{new} \sim H$ and sample $s^{\star\,(i)}_t$ conditional on $y_{1:t}$, and
        $s^{\star\,(i)}_{1:(t-1)}$ from
        $q(k) \propto \begin{cases} m^{(i)}_{k,t-1}\, k\left(y_t \mid \theta^{\star\,(i)}_k\right) & \text{if } 1 \le k \le K^{(i)}_{t-1} \\ M\, k(y_t \mid \theta_{new}) & \text{if } k = K^{(i)}_{t-1}+1 \end{cases}.$
    (b) Calculate the unnormalized weight
        $\xi^{(i)}_t = M k(y_t \mid \theta_{new}) + \sum_{k=1}^{K^{(i)}_{t-1}} m^{(i)}_{k,t-1} k\left(y_t \mid \theta^{\star\,(i)}_k\right).$
  2. Re-weight the particles according to the weights $\zeta_i = \frac{\xi^{(i)}_t}{\sum_{i=1}^N \xi^{(i)}_t}$ $(i = 1, \ldots, N)$.
```

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

The problem of particle degeneracy is most pronounced in Algorithm 2 where static parameters $\theta^\star_{1:K_t}$ are introduced but there is always a potential problem of particle degeneracy in all SMC methods for mixture models since $s^{\star\,(i)}_{1:t}$ act as static parameters when moving beyond the tth iteration. Ulker et al. (2010) suggest sampling a block $s^\star_{(t-r):t}$ conditional on $s^\star_{1:(t-r-1)}$ at the tth iteration to rejuvenate the particles. Alternatively $s^\star_{1:t}$ can be updated in Step 2.

Computational methods for approximating the marginal likelihood $p(y_1, \ldots, y_n)$ are useful in the calculation of Bayes factors for hypothesis testing and can be used in PMCMC methods (Andrieu et al. 2010). Del Moral (2004) shows that the marginal likelihood can be a simply, unbiasedly estimated by

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

which only uses the weights in an SMC sampler.

2.2 Normalized random measures with independent increment mixtures

Bayesian inference for NRMI mixtures was discussed by James et al. (2009). Only the class of homogeneous NRMI will be considered in this paper where

$$G(B)=\frac{\mu(B)}{\mu(\mathbb{Y})}$$

where $\mathbb{Y}$ is the support of G and μ is a completely random measure. If the completely random measure is suitably defined, this implies that

$$\mu=\sum_{k=1}^{\infty}J_k\delta_{\theta_k}$$

where $J_1, J_2, \ldots$ are the jumps of a non-Gaussian Lévy process (*i.e.* a subordinator) with Lévy density $\rho(x)$ and θ is independent of J. In this case, G can be written in the form of (2) with $w_1, w_2, w_3, \ldots$ defined by

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

The process is well-defined if $0<\sum_{l=1}^{\infty}J_l<\infty$ which occurs if $\int_0^{\infty}\rho(x)\,dx=\infty$. The choice of $\rho(x)$ 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 parameter M arises by taking $J_1, J_2, J_3, \ldots$ to be the jumps of a gamma process which has Lévy density $\rho(x)=Mx^{-1}\exp\{-x\}$ (where $M>0$). The normalized generalized gamma (NGG) process (Lijoi et al. 2007) occurs as the normalization of a generalized gamma process (Brix 1999) which has Lévy measure $\rho(x)=\frac{M}{\Gamma(1-\gamma)}x^{-1-\gamma}\exp\{-\lambda x\}$ (where $M>0$, $0<\gamma<1$ and $\lambda>0$). 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 $s_1^\star,\ldots,s_t^\star$ is particularly useful for the conjugate mixture model and can be written

$$\mathrm{pr}(s_{1:t}^\star)=\mathrm{E}\left[\prod_{k=1}^{K_t}w_k^{m_{k,t}}\right].$$

This is referred to as the Exchangeable Product Partition Formula (EPPF) and it only depends on the values of $s_{1:t}^\star$ through $m_{1,t},\ldots,m_{K_t,t}$ and K_t. Following James et al. (2009), it is useful to define the notation

$$\tau_n(u)=\int s^n\exp\{-us\}\rho(s)\,ds$$

and

$$\psi(u)=\int(1-\exp\{-us\})\rho(s)\,ds.$$

James et al. (2009) used the identity $\int_0^{\infty}\exp\{-vx\}\,dv=\frac{1}{x}$ to show that the EPPF can be conveniently written as

$$\begin{aligned}\mathrm{pr}(s_{1:t}^\star)&=\int_0^{\infty}\cdots\int_0^{\infty}\mathrm{E}\left[\prod_{k=1}^{K_t}J_k^{m_{k,t}}\exp\left\{-\sum_{j=1}^{t}v_j\sum_{l=1}^{\infty}J_l\right\}\right]\\&\quad\times dv_1\cdots dv_t\\&=\Gamma(t)^{-1}\int_0^{\infty}\cdots\int_0^{\infty}\exp\{-\psi(U_t)\}\prod_{k=1}^{K_t}\tau_{m_{k,t}}(U_t)\\&\quad\times dv_1\ldots dv_t \qquad (5)\\&=\Gamma(t)^{-1}\int_0^{\infty}U_t^{t-1}\exp\{-\psi(U_t)\}\prod_{k=1}^{K_t}\tau_{m_{k,t}}(U_t)\\&\quad\times dU_t \qquad (6)\end{aligned}$$

where $U_t=\sum_{j=1}^{t}v_j$. This result is particularly important for deriving a tractable expression for the predictive distribution of $s_t^\star$ which can be expressed as

$$\mathrm{pr}\left(s_t^\star\mid s_{1:(t-1)}^\star\right)=\frac{\mathrm{pr}\left(s_{1:t}^\star\right)}{\mathrm{pr}\left(s_{1:(t-1)}^\star\right)}. \qquad (7)$$

In the MCMC literature (Favaro and Teh 2013), it is common to sample $s_{1:t}^\star$ and U_t from the distribution proportional to

$$U_t^{t-1}\exp\{-\psi(U_t)\}\prod_{k=1}^{K_t}\tau_{m_{k,t}}(U_t).$$

The result in (6) implies that the marginal distribution of $\mathrm{pr}\left(s_{1:t}^\star\right)$ with this distribution is the EPPF. The Pólya urn scheme conditional on $v_{1:t}$ is

$$\mathrm{pr}\left(s_t^\star\mid s_{1:(t-1)}^\star,v_{1:t}\right)=\frac{\mathrm{pr}\left(s_{1:t}^\star|U_t\right)}{\mathrm{pr}\left(s_{1:(t-1)}^\star|U_t\right)}. \qquad (8)$$

In the case of the NGG process, this leads to the following expression for the conditional Pólya urn scheme (full details are provided in Sect. 2 of Online Appendix)

$$\begin{aligned}&p\left(s_t^\star=k\mid s_{1:(t-1)}^\star,v_{1:t}\right)\\&\quad=\begin{cases}\frac{m_{k,t-1}-\gamma}{M(\lambda+U_t)^{\gamma}+(t-1)-K_{t-1}\gamma} & \text{if } k\le K_{t-1}\\ \frac{M(\lambda+U_t)^{\gamma}}{M(\lambda+U_t)^{\gamma}+(t-1)-K_{t-1}\gamma} & \text{if } k=K_{t-1}+1\end{cases}.\end{aligned}$$

If $\gamma = 0$ and $\lambda = 1$, this reduces to the Pólya urn scheme familiar from the DP. The probability of joining a previously defined component is proportional to $m_{k,t-1}$ which is the number of observations previously allocated to that component. The probability of joining a new cluster is proportional to M. As γ increases, the probability of allocating to a previously defined component is reduced and the probability of allocating to a new component is increased. This leads to a prior distribution for the number of clusters in a sample of size n which becomes increasingly dispersed.

A second important result derived by James et al. (2009) is the posterior distribution of μ. Let $y_1, \ldots, y_t$ be independent and identically distributed according to G then the posterior of μ conditional on U_t 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 kth distinct value of $y_1, \ldots, y_t$ and $p(\hat{J}_k \mid y) \propto \rho(\hat{J}_k)\hat{J}_k^{m_{k,t}} \exp\{-\hat{J}_k U_t\}$ and $(\tilde{J}, \tilde{\theta})$ where $\tilde{J}$ is a Poisson process with intensity $\rho(J)\exp\{-JU_t\}$ and $\tilde{\theta}_k \overset{i.i.d.}{\sim} H \quad (k = 1, 2, \ldots)$.

3 Sequential Monte Carlo methods for NRMI mixtures

3.1 Conjugate NRMI mixtures

An SMC algorithm for conjugate NRMI mixture models could be defined by extending the methods for DP mixtures described in Sect. 2.1. An expression for the conditional distribution of $s_t^\star$ given $s_{1:(t-1)}^\star$ and $v_{1:t}$ for any NRMI mixture is available using (7). This is a finite, discrete distribution but it can be difficult to compute the probabilities of different values of $s_t^\star$ for many choices of $\rho(x)$. Therefore, the proposed SMC algorithm for conjugate NRMI mixtures uses the extended state $(s_t^\star, v_t)$ whose joint prior distribution is

$$\mathrm{pr}(s_{1:t}^\star, v_{1:t}) = \Gamma(t)^{-1} \exp\left\{-\psi\left(U_t\right)\right\} \prod_{k=1}^{K_t} \tau_{m_{k,t}}\left(U_t\right). \tag{9}$$

The predictive distribution of $s_t^\star$ and v_t can be expressed as

$$\begin{aligned}\mathrm{pr}&\left(s_t^\star, v_t \mid s_{1:(t-1)}^\star, v_{1:(t-1)}\right)\\ &= \mathrm{pr}\left(s_t^\star \mid s_{1:(t-1)}^\star, v_{1:t}\right) \mathrm{pr}\left(v_t \mid s_{1:(t-1)}^\star, v_{1:(t-1)}\right)\end{aligned}$$

where

$$\mathrm{pr}\left(v_t \mid s_{1:(t-1)}^\star, v_{1:(t-1)}\right) = \frac{\mathrm{pr}\left(s_{1:(t-1)}^\star, v_{1:t}\right)}{\mathrm{pr}\left(s_{1:(t-1)}^\star, v_{1:(t-1)}\right)}$$

and

$$\mathrm{pr}\left(s_t^\star \mid s_{1:(t-1)}^\star, v_{1:t}\right) = \frac{\mathrm{pr}\left(s_{1:t}^\star, v_{1:t}\right)}{\mathrm{pr}\left(s_{1:(t-1)}^\star, v_{1:t}\right)}. \tag{10}$$

It follows from (6) and (9) that

$$\begin{aligned}\mathrm{pr}&\left(s_{1:(t-1)}^\star, v_{1:t}\right)\\ &= \sum_{s_t^\star=1}^{K_{t-1}+1} \mathrm{pr}\left(s_{1:t}^\star, v_{1:t}\right)\\ &\quad \times \sum_{s_t^\star=1}^{K_{t-1}+1} \mathrm{E}\left[\prod_{k=1}^{K_t} J_k^{m_{k,t}} \exp\left\{-U_t \sum_{l=1}^{\infty} J_l\right\}\right]\\ &= \mathrm{E}\left[\sum_{l=1}^{\infty} J_l \prod_{k=1}^{K_{t-1}} J_k^{m_{k,t-1}} \exp\left\{-U_t \sum_{l=1}^{\infty} J_l\right\}\right]\\ &= -\mathrm{E}\left[\prod_{k=1}^{K_{t-1}} J_k^{m_{k,t-1}} \frac{d}{dv_t} \exp\left\{-U_t \sum_{l=1}^{\infty} J_l\right\}\right]\\ &= -\frac{d}{dv_t}\left[\exp\left\{-\psi\left(U_t\right)\right\} \prod_{k=1}^{K_{t-1}} \tau_{m_{k,t-1}}\left(U_t\right)\right].\end{aligned}$$

This implies that

$$\begin{aligned}\mathrm{pr}&\left(v_t \mid v_{1:(t-1)}, s_{1:(t-1)}^\star\right) \propto\\ &-\frac{d}{dv_t}\left[\prod_{k=1}^{K_{t-1}} \tau_{m_{k,t-1}}\left(U_t\right) \exp\left\{-\psi\left(U_t\right)\right\}\right]\end{aligned}$$

and, clearly, its distribution function is

$$\frac{\prod_{k=1}^{K_{t-1}} \tau_{m_{k,t-1}}\left(U_t\right) \exp\left\{-\psi\left(U_t\right)\right\}}{\prod_{k=1}^{K_{t-1}} \tau_{m_{k,t-1}}\left(U_{t-1}\right) \exp\left\{-\psi\left(U_t\right)\right\}}.$$

Values of v_t can always be simulated using inversion sampling. The conditional distribution of $s_t^\star$ is

$$\mathrm{pr}\left(s_t^\star = k \mid s_{1:(t-1)}^\star, v_{1:t}\right) = \frac{\mathrm{pr}\left(s_{1:(t-1)}^\star, s_t^\star = k, v_{1:t}\right)}{p\left(s_{1:(t-1)}^\star, v_{1:t}\right)}$$

which is a finite, discrete distribution and so can be sampled easily. The full algorithm for the conjugate NRMI mixture model is shown in Algorithm 3. Unlike the algorithm for DP mixtures described in Sect. 2.1, an adaptive resampling step is introduced which can lead to more accurate estimates from the SMC method than resampling at every step (see e.g. Del Moral et al. 2006). The resampling step uses the effective sample size (ESS) which can be loosely interpreted as the number of independent samples needed to produce estimates with the same Monte Carlo error as the SMC algorithm. In this (and subsequent algorithms), resampling only occurs if the ESS is below some threshold aN (where $a = 0.5$ is a standard value used throughout the SMC literature and in this paper). Posterior summaries are calculated as weighted average so that e.g.

$$\mathrm{E}[f(s^{\star}_{1:t})|y_{1:t}] = \frac{\sum_{i=1}^{N} \xi_t^{(i)} f\left(s_{1:t}^{\star(i)}\right)}{\sum_{i=1}^{N} \xi_t^{(i)}}.$$

```
Choose a threshold $a$ $(0 < a < 1)$, initialize $\xi_0^{(1)} = 1, \ldots, \xi_0^{(N)} = 1$.
For $t = 1, \ldots, n$, perform steps (1)–(3)
  1. For $i = 1, \ldots, N$ perform steps (a)–(c)
     (a) Sample $v_t^{(i)}$ from the distribution $v_t^{(i)} \mid s^{\star(i)}_{1:(t-1)}, v^{(i)}_{1:(t-1)}$.
     (b) Sample $s_t^{\star(i)}$ from the probability mass function
         $q(k) \propto \begin{cases} \mathrm{pr}\left(s_t^{\star(i)} = k \mid s^{\star(i)}_{1:(t-1)}, v^{(i)}_{1:t}\right) k_k^{\star}\left(y_t \mid s^{\star(i)}_{1:(t-1)}\right) \\ \quad \text{if } k \le K^{(i)}_{t-1} \\ \mathrm{pr}\left(s_t^{\star(i)} = K^{(i)}_{t-1} + 1 | s^{\star(i)}_{1:(t-1)}, v^{(i)}_{1:t}\right) k_{new}\left(y_t\right) \\ \quad \text{if } k = K^{(i)}_{t-1} + 1 \end{cases}.$
     (c) Update the unnormalized weight
         $\xi_t^{(i)} = \xi_{t-1}^{(i)}\Big[\mathrm{pr}\left(s_t^{\star(i)} = K^{(i)}_{t-1} + 1 \mid s^{\star(i)}_{1:(t-1)}, v^{(i)}_{1:t}\right) k_{new}\left(y_t\right)$
         $+ \sum_{k=1}^{K^{(i)}_{t-1}} \mathrm{pr}\left(s_t^{\star(i)} = k \mid s^{\star(i)}_{1:(t-1)}, v^{(i)}_{1:t}\right) k_k\left(y_t \mid s^{\star(i)}_{1:(t-1)}\right)\Big].$
  2. Calculate ESS $= \frac{\left(\sum_{i=1}^{N} \xi_t^{(i)}\right)^2}{\sum_{i=1}^{N} \xi_t^{(i)\,2}}$.
  3. If ESS $< aN$, then re-weight the particles according to the weights $\zeta_i = \frac{\xi_t^{(i)}}{\sum_{i=1}^{N} \xi_t^{(i)}}$ $(i = 1, \ldots, N)$ and update $s^{\star(i)}_{1:t}$
     $(i = 1, \ldots, N)$ using MCMC. Set $\xi_t^{(1)} = 1, \ldots, \xi_t^{(N)} = 1$.
```

Algorithm 3: SMC algorithm for conjugate NRMI mixture models

3.2 Non-conjugate NRMI mixtures

The algorithm defined in the previous subsection exploit the conjugacy of the mixture model and some properties of NRMI priors to define an algorithm that works directly on the allocation variables $s^{\star}_{1:t}$ and $v_{1:t}$. Non-conjugate nonparametric mixture models typically lead to additional computational effort since the random measure cannot be analytically integrated from the model.

Two SMC methods for non-conjugate NRMI mixture models will be considered. The first directly extends the samplers defined in Sect. 3.1 by integrating out the sizes of the jumps, $J_1, J_2, \ldots$ and so extends Favaro and Teh (2013) from MCMC to SMC. The second extends slice sampling methods for NRMI mixture models (Griffin and Walker 2011) from MCMC to SMC.

The first method for non-conjugate mixture models is defined in the spirit of Favaro and Teh (2013) who extend Algorithm 8 of Neal (2000) for DP mixture models to NRMI mixtures by including U_t as an auxiliary variable in an MCMC framework. Algorithm 2 can be extended by sampling m values $\theta_{new,1}, \ldots, \theta_{new,m} \overset{i.i.d.}{\sim} H_t^{\star}$ in Step 1(a) in place of θ_{new}. The algorithm allows for values of θ_{new} drawn from a distribution $H_t^{\star}$ which can be chosen to reflect the centring measure H and y_t. The choice $H_t^{\star} = H$ leads to a simplification of the proposal distribution for $s_t^{(i)}$ and the weight. The auxiliary particle filter (Pitt and Shephard 1999) would choose $h_t^{\star}\left(\tilde{\theta}_k^{(i)}\right) \propto h\left(\tilde{\theta}_k^{(i)}\right) k\left(y_t|\tilde{\theta}_k^{(i)}\right)$. If this choice cannot be sampled straightforwardly then a choice of $h_t^{\star}\left(\tilde{\theta}_k^{(i)}\right)$ that approximates this distribution could be used. The full algorithm is presented as Algorithm 4. Step 1(a) can be completed using the methods developed for conjugate NRMI mixture models and the updating in Step 3 can be completed using the MCMC methods described in Favaro and Teh (2013).

```
Choose a threshold $a$ $(0 < a < 1)$, initialize $\xi_0^{(1)} = 1, \ldots, \xi_0^{(N)} = 1$.
For $t = 1, \ldots, n$, perform steps (1)–(3)
  1. For $i = 1, \ldots, N$ perform steps (a)–(d)
     (a) Sample $v_t^{(i)}$ from the distribution $v_t^{(i)} \mid s^{\star(i)}_{1:(t-1)}, v^{(i)}_{1:(t-1)}$.
     (b) Sample $\theta_{new,1:m} \overset{i.i.d.}{\sim} H_t^{\star}$
     (c) Sample $s_t^{\star(i)}$ from the probability mass function
         $q(k) \propto \begin{cases} \mathrm{pr}\left(s_t^{\star(i)} = k \mid s^{\star(i)}_{1:(t-1)}, v^{(i)}_{1:t}\right) k_k^{\star}\left(y_t \mid \theta_k^{\star(i)}\right) \\ \quad \text{if } k \le K^{(i)}_{t-1} \\ \mathrm{pr}\left(s_t^{\star(i)} = K^{(i)}_{t-1} + 1 | s^{\star(i)}_{1:(t-1)}, v^{(i)}_{1:t}\right) \\ \quad \frac{1}{m}\sum_{j=1}^{m} k\left(y_t \mid \theta_{new,j}\right) \frac{h(\theta_{new,j})}{h_t^{\star}(\theta_{new,j})} \text{if } k = K^{(i)}_{t-1} + 1 \end{cases}.$
         If $s_t^{\star(i)} = K^{(i)}_{t-1} + 1$, then set $\theta^{\star(i)}_{K^{(i)}_{t-1}+1} = \theta_{new,j}$ with probability $\frac{k(y_t|\theta_{new,j}) \frac{h(\theta_{new,j})}{h_t^{\star}(\theta_{new,j})}}{\sum_{l=1}^{m} k(y_t|\theta_{new,l}) \frac{h(\theta_{new,l})}{h_t^{\star}(\theta_{new,l})}}$.
     (d) Update the unnormalized weight
         $\xi_t^{(i)} = \xi_{t-1}^{(i)}\Big[\mathrm{pr}\left(s_t^{\star(i)} = K^{(i)}_{t-1} + 1 | s^{\star(i)}_{1:(t-1)}, v^{(i)}_{1:t}\right)$
         $\frac{1}{m}\sum_{j=1}^{m} k\left(y_t \mid \theta_{new,j}\right) \frac{h(\theta_{new,j})}{h_t^{\star}(\theta_{new,j})}$
         $+ \sum_{k=1}^{K^{(i)}_{t-1}} \mathrm{pr}\left(s_t^{\star(i)} = k \mid s^{\star(i)}_{1:(t-1)}, v^{(i)}_{1:t}\right) k_k\left(y_t \mid \theta_k^{\star(i)}\right)\Big].$
  2. Calculate ESS $= \frac{\left(\sum_{i=1}^{N} \xi_t^{(i)}\right)^2}{\sum_{i=1}^{N} \xi_t^{(i)\,2}}$.
  3. If ESS $< aN$, then re-weight the particles according to the weights $\zeta_i = \frac{\xi_t^{(i)}}{\sum_{i=1}^{N} \xi_t^{(i)}}$ $(i = 1, \ldots, N)$ and update $s^{\star(i)}_{1:t}, \theta^{\star(i)}_{1:K_t^{(i)}}$
     $(i = 1, \ldots, N)$ using MCMC. Set $\xi_t^{(1)} = 1, \ldots, \xi_t^{(N)} = 1$.
```

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

The second method is based on slice sampling. Slice samplers are auxiliary variable MCMC methods which introduce latent variables that make all steps of the Gibbs sampler involve only a finite number of the distinct values of G. Griffin and Walker (2011) described two Gibbs samplers which efficiently simulate from any NRMI mixture models without truncation error. They define their Slice 1 sampler using the allocation variables $s_1, \ldots, s_n$ (rather than $s_1^\star, \ldots, s_n^\star$ used in the previous section) by writing the likelihood contribution $\prod_{j=1}^t w_{s_j} k(y_j \mid \theta_{s_j})$ in the following way

$$\prod_{j=1}^{t} \mathrm{I}(\kappa_j < J_{s_j}) k(y_j \mid \theta_{s_j}) \exp\left\{-v_j \sum_{k=1}^{\infty} J_k\right\} \tag{11}$$

where $\mathrm{I}(\cdot)$ is the indicator function. Integrating out $v_1, \ldots, v_t$ and $\kappa_1, \ldots, \kappa_t$ leads to the original likelihood contribution. They defined their Slice 2 sampler by writing the likelihood contribution in the alternative form

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

where $\alpha_t = \min\{J_{s_j} \mid j = 1, \ldots, t\}$. The introduction of the latent variables $\kappa_1, \ldots, \kappa_t$ in Slice 1 and κ in Slice 2 leads to likelihood contributions that only depend on a finite number of jumps and locations. A finite dimensional representation of the posterior which is suitable for simulation can be defined by integrating out all other jumps and locations.

The forms of the likelihood introduced in Slice 1 and Slice 2 are also convenient for SMC methods since the number of latent parameters grows with the number of observations. In addition to s_t, there are states κ_t and v_t in sampler 1 and v_t in sampler 2 (with κ treated as a static parameter). However, it is not immediately clear how to sample from their joint predictive distributions. 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 Sect. 3) from the model then the latent variable v_t is sampled using the method for a conjugate model. The latent variable κ_t is sampled by first simulating another latent variable d_t according to the conditional distribution of s_t given in (10). If d_t is associated with a new jump then a new value is drawn from the centring distribution H and added to $\hat{\theta}$. A random variable ν_t is introduced with $\nu_t = 1$ if d_t is associated with a new jump and $\nu_t = 0$ otherwise. The points in $\hat{J}$ are then simulated conditional on $s_{1:(t-1)}$ and ν_t and associated with $\hat{\theta}$. Finally, κ_t is simulated from $\mathrm{U}\left(0, \hat{J}_{d_t}\right)$. This allows us to simulate the R_t jumps with size in (κ_t, ∞) and no observation allocated. These are denoted $\tilde{J}_1, \ldots, \tilde{J}_{R_t}$ which follow a Poisson process with intensity $\exp\{-J\ U_t\}\,\rho(J)$. Values of $\tilde{\theta}$ are simulated from H and associated with each point of $\tilde{J}$. The sample of κ_t, $\hat{J}$ and $\tilde{J}$ are from the joint distribution of κ_t and J (restricted to (κ_t, ∞)) conditional on previous values. This allows us to sample s_t from its conditional distribution defined by (11).

Once all particles have been sampled, they are reweighted. Algorithm 5 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 κ_t in Slice 1 can be replaced by the following sampling step for κ, simulated according to $\kappa \sim \mathrm{U}(0, \beta_t)$ 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 $\exp\{-J U_t\}\rho(J)$ restricted to the interval (κ, ∞). 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 6 describes the full method.

3.3 Estimating hyperparameters

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 SMC 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. Alternatively, the parameters could be integrated out from the model. This paper adopts the alternative method of updating the static parameters using a Gibbs step when the particles are resampled.

4 Particle Markov chain Monte Carlo methods

Particle Markov chain Monte Carlo methods (Andrieu et al. 2010) use SMC methods in MCMC algorithms for static inference. There are two main cases of method: particle Metropolis–Hastings and particle Gibbs sampler. Consider the model in (4). The marginal posterior distribution of ϕ can be sampled using a particle Metropolis–Hastings method. An SMC method is used to unbiasedly estimate the marginal likelihood $p(y|\phi)$. This estimate is used in place of the actual marginal likelihood in the usual Metropolis–Hastings sampler. Algorithm 7 gives further details and Andrieu et al. (2010) show that this sampler produce draws from the posterior distribution of ϕ.

In order to perform cluster or density estimation, a posterior sample from $s_{1:n}^\star$ is needed and so I will concentrate on particle Gibbs methods which can produce such a sample. Particle Gibbs methods use an SMC algorithm to jointly update states in a Gibbs sampler. In MCMC samplers where the Pólya urn scheme representation is used such as methods for conjugate mixtures and auxiliary variable samplers (Favaro and Teh 2013), particle Gibbs methods can be used to jointly update the allocations $s^{\star}_{1:n}$. In so-called conditional methods, the allocations are jointly updated conditional on some jumps of the mixing measure (e.g. Papaspiliopoulos and Roberts 2008; Kalli et al. 2011; Griffin and Walker 2011) and so particle Gibbs methods offer no benefit.

It is assumed that the posterior distribution of $s^{\star}$ and ϕ for the model in (4) is to be sampled using a Gibbs sampler. In particle Gibbs sampling, ϕ is updated from its full conditional distribution and $s^{\star}$ is updated using a conditional particle filter (Andrieu et al. 2010) which uses the current value of $s^{\star}$ as a reference trajectory in an SMC algorithm. A conditional particle filter for a conjugate NRMI mixture model is described in Algorithm 8. The first particle is the reference trajectory which is fixed in the particle filter. Otherwise, the algorithm evolves according to Algorithm 3 with new states proposed and weights calculated (for all states including the reference trajectory). This algorithm generalizes the original CPF method by allowing different re-weighting schemes (as discussed by Chopin and Singh 2013) and using adaptive resampling (e.g. Andrieu et al. 2010). The basic algorithm of Andrieu et al. (2010) arises if $a = 1$ and the particles are re-weighted using multinomial sampling. A full description of extension to stratified and residual resampling schemes is given by Chopin and Singh (2013). Other variations on the conditional particle filter have been proposed including backward sampling (Whiteley 2010; Whiteley et al. 2010) and updating of the trajectory in the SMC algorithm (Lindsten et al. 2014).

```
Choose a threshold $a$ $(0 < a < 1)$, initialize $U_0^{(1)} = 0, \ldots, U_0^{(N)} = 0$ and $\xi_0^{(1)} = 1, \ldots, \xi_0^{(N)} = 1$. For $t = 1, \ldots, n$, perform steps (1)–(3)
  1. For $i = 1, \ldots, N$, perform steps (a)–(g)
     (a) Sample $v_t^{(i)}$ from the distribution $v_t^{(i)} \mid s^{(i)}_{1:(t-1)}, v^{(i)}_{1:(t-1)}$ and set $U_t^{(i)} = U_{t-1}^{(i)} + v_t^{(i)}$.
     (b) Sample $d_t^{(i)}$ from the distribution proportional to $p\left(s_t^{(i)} \mid s^{(i)}_{1:(t-1)}, v^{(i)}_{1:t}\right)$, If $d_t^{(i)} = K_{t-1}^{(i)} + 1$, simulate $\tilde{\theta}^{(i)}_{K_{t-1}^{(i)}+1} \sim H_t^{\star}$. Set $K_t^{\star(i)} = K_{t-1}^{(i)} + 1$ if $d_t^{(i)} = K_{t-1}^{(i)} + 1$ and $K_t^{\star(i)} = K_{t-1}^{(i)}$ otherwise.
     (c) Sample $\hat{J}_1^{(i)}, \ldots, \hat{J}^{(i)}_{K_t^{\star(i)}}$. The density of $\hat{J}_k^{(i)}$ is proportional to
         $\left(\hat{J}_k^{(i)}\right)^{m^{(i)}_{k,t-1} + \mathrm{I}\left(d_t^{(i)} = k\right)} \exp\left\{-\hat{J}_k^{(i)} U_t^{(i)}\right\} \rho\left(\hat{J}_k^{(i)}\right).$
     (d) Sample $\kappa_t^{(i)} \sim \mathrm{U}\left(0, \hat{J}^{(i)}_{d_t^{(i)}}\right)$
     (e) Sample $\tilde{J}_1^{(i)}, \ldots, \tilde{J}^{(i)}_{R_t^{(i)}}$ from a Poisson process on $\left(\kappa_t^{(i)}, \infty\right)$ with intensity $\exp\left\{-J U_t^{(i)}\right\} \rho(J)$. Simulate $\tilde{\theta}_1^{(i)}, \ldots, \tilde{\theta}^{(i)}_{R_t^{(i)}} \overset{i.i.d.}{\sim} H_t^{\star}$.
     (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
         $p(s_t^{(i)} = k) \propto \begin{cases} \mathrm{I}\left(\hat{J}_k > \kappa_t^{(i)}\right) k\left(y_t \mid \hat{\theta}_k^{(i)}\right) & 1 \le k \le K_t^{\star(i)} \\ \mathrm{I}\left(\tilde{J}_k > \kappa_t^{(i)}\right) k\left(y_t \mid \tilde{\theta}_k^{(i)}\right) \frac{h(\theta_k^{(i)})}{h_t^{\star}(\theta_k^{(i)})} & K_t^{\star(i)} + 1 \le k \le K_t^{\star(i)} + R_t^{(i)} \end{cases}.$
     (g) Update the unnormalized weight
         $\xi_t^{(i)} = \xi_{t-1}^{(i)} \frac{\sum \mathrm{I}\left(\hat{J}_k^{(i)} > \kappa_t^{(i)}\right) k\left(y_t \mid \hat{\theta}_k^{(i)}\right) + \sum \mathrm{I}\left(\tilde{J}_k^{(i)} > \kappa_t^{(i)}\right) k\left(y_t \mid \tilde{\theta}_k^{(i)}\right) \frac{h(\tilde{\theta}_k^{(i)})}{h_t^{\star}(\tilde{\theta}_k^{(i)})}}{\sum \mathrm{I}\left(\hat{J}_k^{(i)} > \kappa_t^{(i)}\right) + \sum \mathrm{I}\left(\tilde{J}_k^{(i)} > \kappa_t^{(i)}\right)}.$
  2. Calculate ESS $= \frac{\left(\sum_{i=1}^{N} \xi_t^{(i)}\right)^2}{\sum_{i=1}^{N} \xi_t^{(i)2}}$.
  3. If ESS $< aN$, then re-weight the particles according to the weights $\zeta_i = \frac{\xi_t^{(i)}}{\sum_{i=1}^{N} \xi_t^{(i)}}$ $(i = 1, \ldots, N)$ and update all parameters using MCMC.
     Set $\xi_t^{(1)} = 1, \ldots, \xi_t^{(N)} = 1$.
```

Algorithm 5: Slice 1 SMC algorithm for non-conjugate NRMI mixture models

5 Illustrations

5.1 Comparison of SMC methods

The infinite mixture of normals model is one of the most popular in Bayesian nonparametrics and was a natural testing

```
Choose a threshold $a$ $(0 < a < 1)$, initialize $U_0^{(1)} = 0, \ldots, U_0^{(N)} = 0$ and $\xi_0^{(1)} = 1, \ldots, \xi_0^{(N)} = 1$. For $t = 1, \ldots, n$, perform steps (1)–(3)
  1. For $i = 1, \ldots, N$, perform steps (a)–(h)
      (a) Sample $v_t^{(i)}$ from the distribution $v_t^{(i)} \mid s_{1:(t-1)}^{(i)}, v_{1:(t-1)}^{(i)}$ and set $U_t^{(i)} = U_{t-1}^{(i)} + v_t^{(i)}$.
      (b) Sample $d_t^{(i)}$ from the distribution proportional to $p\left(s_t^{(i)} \mid s_{1:(t-1)}^{(i)}, v_{1:t}^{(i)}\right)$. If $d_t^{(i)} = K_{t-1}^{(i)} + 1$, simulate $\tilde{\theta}_{K_{t-1}^{(i)}+1}^{(i)} \sim H_t^\star$. Set $K_t^{\star(i)} = K_{t-1}^{(i)} + 1$ if $d_t^{(i)} = K_{t-1}^{(i)} + 1$ and $K_t^{\star(i)} = K_{t-1}^{(i)}$ otherwise.
      (c) Sample $\hat{J}_1^{(i)}, \ldots, \hat{J}_{K_t^{\star(i)}}^{(i)}$. The density of $\hat{J}_k^{(i)}$ is proportional to
          $$\left(\hat{J}_k^{(i)}\right)^{m_{k,t-1}^{(i)} + I\left(d_t^{(i)}=k\right)} \exp\left\{-\hat{J}_k^{(i)} U_t^{(i)}\right\} \rho\left(\hat{J}^{(i)}\right).$$
      (d) Let $\alpha_{t-1}^{(i)} = \min\left\{\hat{J}_1^{(i)}, \ldots, \hat{J}_{K_{t-1}^{(i)}}^{(i)}\right\}$ and $\beta_t^{(i)} = \min\left\{\hat{J}_1^{(i)}, \ldots, \hat{J}_{K_t^{\star(i)}}^{(i)}\right\}$.
      (e) Sample $\kappa^{(i)} \sim \mathrm{U}\left(0, \beta_t^{(i)}\right)$
      (f) Sample $\tilde{J}_1^{(i)}, \ldots, \tilde{J}_{R_t^{(i)}}^{(i)}$ from a Poisson process on $\left(\kappa^{(i)}, \infty\right)$ with intensity $\exp\left\{-J U_t^{(i)}\right\} \rho(J)$. Simulate $\tilde{\theta}_1^{(i)}, \ldots, \tilde{\theta}_{R_t^{(i)}}^{(i)} \overset{i.i.d.}{\sim} H_t^\star$.
      (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
          $$q\left(s_t^{(i)} = k\right) \propto \begin{cases} \max\left\{\hat{J}_k^{(i)}, \alpha_{t-1}^{(i)}\right\} k\left(y_t \mid \hat{\theta}_k^{(i)}\right) & 1 \le k \le K_t^{\star(i)} \\ \max\left\{\tilde{J}_k^{(i)}, \alpha_{t-1}^{(i)}\right\} k\left(y_t \mid \tilde{\theta}_k^{(i)}\right) \frac{h(\tilde{\theta}_k^{(i)})}{h_t^\star(\tilde{\theta}_k^{(i)})} & K_t^{\star(i)} + 1 \le k \le K_t^{\star(i)} + R_t^{(i)} \end{cases}.$$
      (h) Update the unnormalized weight
          $$\xi_t^{(i)} = \xi_{t-1}^{(i)} \frac{\sum \max\left\{\hat{J}_k^{(i)}, \alpha_{t-1}^{(i)}\right\} k\left(y_t \mid \hat{\theta}_k^{(i)}\right) + \sum \max\left\{\tilde{J}_k^{(i)}, \alpha_{t-1}^{(i)}\right\} k\left(y_t \mid \tilde{\theta}_k^{(i)}\right) \frac{h(\tilde{\theta}_k^{(i)})}{h_t^\star(\tilde{\theta}_k^{(i)})}}{\sum \max\left\{\hat{J}_k^{(i)}, \alpha_{t-1}^{(i)}\right\} + \sum \max\left\{\tilde{J}_k^{(i)}, \alpha_{t-1}^{(i)}\right\}}$$
  2. Calculate ESS $= \frac{\left(\sum_{i=1}^N \xi_t^{(i)}\right)^2}{\sum_{i=1}^N \xi_t^{(i)2}}$.
  3. If ESS $< aN$, then re-weight the particles according to the weights $\zeta_i = \frac{\xi_t^{(i)}}{\sum_{i=1}^N \xi_t^{(i)}}$ $(i = 1, \ldots, N)$ and update all parameters using MCMC.
     Set $\xi_t^{(1)} = 1, \ldots, \xi_t^{(N)} = 1$.
```

Algorithm 6: Slice 2 SMC algorithm for non-conjugate NRMI mixture models

ground for the methods developed in this paper. The infinite mixture model introduced by Griffin (2010) was used

$$
\begin{aligned}
y_t|\mu_t &\sim \mathrm{N}(\mu_t, a\sigma^2), \quad t = 1, \ldots, n \\
\mu_t &\sim G, \quad t = 1, \ldots, n \\
G &\sim \mathrm{NGG}(\gamma, 1, M, H)
\end{aligned}
\tag{12}
$$

where H is a normal distribution with mean μ_0 and variance $(1 - a)\sigma^2$. Inference for the posterior distribution on the full sample $y_1, \ldots, y_n$ only was considered to allow comparsion to MCMC methods. SMC methods for this model were applied to two datasets: the ever-popular galaxy data and the log acidity data. The data were standardized to have mean 0 and variance 1 and the parameter values $\mu_0 = 0$ and $\sigma = 1$ were chosen. The parameter a was 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 SMC algorithms was run with 5000 particles.

```
1. Initialize $\phi$ and calculate an approximation $L$ of $p(y|\phi)$ using an SMC sampler.
2. Suppose that the current value of the parameter is $\phi$ and that a new value $\phi'$ is proposed according to the conditional density $q\left(\phi'|\phi\right)$. Calculate an approximation $L'$ of $p(y|\phi')$ using an SMC sampler and
       $$\alpha\left(\phi, \phi'\right) = \frac{L' p(\phi') q\left(\phi|\phi'\right)}{L p\left(\phi\right) q\left(\phi'|\phi\right)}.$$
   If $\alpha\left(\phi, \phi'\right) < u$, accept $(\phi', L')$, otherwise retain $(\phi, L)$.
3. Return to Step 2.
```

Algorithm 7: General particle Metropolis-Hastings algorithm CPF algorithm for an NRMI mixture models

Initially, a comparison of the methods for conjugate Dirichlet process mixture model was performed. The methods considered were Algorithms 3, 4 (with $m = 3, 27, 250$) and 5. The number of clusters was used as the parameter of interest to calculate the effective sample size (ESS) using the method of Carpenter et al. (1999). They assumed that the posterior expectation to be approximated was $\xi = \mathrm{E}[g(\eta)|y_{1:t}]$ where η were parameters of the model being estimated and that R runs of the SMC method were performed. If the estimate of ξ on the rth run was $z_r = \sum_{i=1}^{N} \zeta_r^{(i)} g\left(\eta_r^{(i)}\right)$ and $v_r = \sum_{i=1}^{N} \zeta_r^{(k)} g\left(\eta_r^{(i)}\right)^2 - z_r^2$, the ESS was estimated by $\frac{M\bar{v}}{\sum_{r=1}^{R}(z_r - \bar{z})^2}$ where $\bar{v}$ and $\bar{z}$ were the sample means of $v_1, \ldots, v_R$ and $z_1, \ldots, z_R$ respectively. The computational time was calculated using the "tic-toc" function of Matlab. The relative efficiency (R.E.) was defined to be the ratio of the ESS and the computational time and so represented the effective number of samples per unit of computational time.

```
Choose a threshold $a$ $(0 < a < 1)$, initialize $\xi_0^{(1)} = 1, \ldots, \xi_0^{(N)} = 1$.
Set the reference trajectory to be $s_{1:n}^{\star(1)}$ and $v_{1:n}^{(1)}$.

1. For $t = 1, \ldots, n$,
   (a) For $i = 2, \ldots, N$,
       i. Sample $v_t^{(i)}$ from the distribution $v_t^{(i)} \mid s_{1:(t-1)}^{\star(i)}, v_{1:(t-1)}^{(i)}$.
       ii. Sample $s_t^{\star(i)}$ from the probability mass function
           $q(k) \propto \begin{cases} \mathrm{pr}\left(s_t^{\star(i)} = k \mid s_{1:(t-1)}^{\star(i)}, v_{1:t}^{(i)}\right) \\ k_k^{\star}\left(y_t \mid s_{1:(t-1)}^{\star(i)}\right) & \text{if } k \le K_{t-1}^{(i)} \\ \mathrm{pr}\left(s_t^{\star(i)} = K_{t-1}^{(i)} + 1 | s_{1:(t-1)}^{\star(i)}, v_{1:t}^{(i)}\right) \\ k_{new}(y_t) & \text{if } k = K_{t-1}^{(i)} + 1 \end{cases}$.
   (b) For $i = 1, \ldots, N$, update the unnormalized weight
       $\xi_t^{(i)} = \xi_{t-1}^{(i)}\Big[\mathrm{pr}\left(s_t^{\star(i)} = K_{t-1}^{(i)} + 1 \mid s_{1:(t-1)}^{\star(i)}, v_{1:t}^{(i)}\right) k_{new}(y_t)$
       $+ \sum_{k=1}^{K_{t-1}^{(i)}} \mathrm{pr}\left(s_t^{\star(i)} = k \mid s_{1:(t-1)}^{\star(i)}, v_{1:t}^{(i)}\right) k_k\left(y_t \mid s_{1:(t-1)}^{\star(i)}\right)\Big]$.
   (c) Calculate ESS $= \frac{\left(\sum_{i=1}^{N} \xi_t^{(i)}\right)^2}{\sum_{i=1}^{N} \xi_t^{(i)2}}$.
   (d) If ESS $< aN$, then re-weight the particles according to the
       weights $\zeta_i = \frac{\xi_t^{(i)}}{\sum_{i=1}^{N} \xi_t^{(i)}}$ $(i = 1, \ldots, N)$ and update $s_{1:t}^{\star(i)}$
       $(i = 1, \ldots, N)$ using MCMC. Set $\xi_t^{(1)} = 1, \ldots, \xi_t^{(N)} = 1$.
2. Simulate $j$ from the distribution with probability mass function
   $p(j) = \frac{\zeta_j}{\sum_{k=1}^{N} \zeta_k}$, $j = 1, \ldots, N$. Return $s_{1:n}^{\star(j)}$ and $v_{1:n}^{\star(j)}$.
```

Algorithm 8: General CPF algorithm for conjugate NRMI mixture models

Table 1 The ESS of estimating the posterior mean number of clusters from 5000 particles with a DP mixture model

Algorithm	Galaxy			Log acidity		
	ESS	Time	R.E.	ESS	Time	R.E.
Algorithm 3	3048	32	95.3	2233	81	27.6
Algorithm 4 ($m = 3$)	736	85	8.7	536	210	2.6
Algorithm 4 ($m = 27$)	2004	90	22.3	1373	224	6.1
Algorithm 4 ($m = 250$)	2136	103	22.5	1660	257	6.5
Algorithm 5	593	142	4.2	428	327	1.3

Table 2 The ESS of estimating the posterior mean number of clusters from 5000 particles with an NGG process prior

Algorithm	Galaxy			Log acidity		
	ESS	Time	R.E.	ESS	Time	R.E.
Algorithm 3	3349	334	10.0	1747	581	3.0
Algorithm 4 ($m = 3$)	389	378	1.0	321	739	0.4
Algorithm 4 ($m = 27$)	1311	370	3.5	930	738	1.3
Algorithm 4 ($m = 250$)	2188	427	5.1	1009	811	1.2
Algorithm 5	625	531	1.0	511	1258	0.4

Table 3 The ESS of estimating the posterior mean number of a from 5000 particles with a DP mixture model

Algorithm	Galaxy			Log acidity		
	ESS	Time	R.E.	ESS	Time	R.E.
Algorithm 3	1107	45	24.6	814	81	10.0
Algorithm 4 ($m = 3$)	38	103	0.4	190	204	0.9
Algorithm 4 ($m = 27$)	190	104	1.8	385	207	1.9
Algorithm 4 ($m = 250$)	382	127	3.0	274	217	1.3
Algorithm 5	170	144	1.2	302	270	1.1

Results are presented in Table 1. Algorithm 3 gave the largest ESS for both data sets and was used as a benchmark against which the non-conjugate samplers (Algorithms 4 and 5) which do not exploit the conjugacy of the mixture model were compared. Algorithm 4 outperformed Algorithm 5 for both data sets. The value of m in Algorithm 4 had a substantial effect on the ESS which was roughly three times larger for $m = 250$ compared to $m = 3$ and was much closer to the ESS for Algorithm 3. The effect on average computational time of increasing m is small and so large values of m are preferable.

Results for the same model with an NGG process prior with $\gamma = 0.2$ for the mixing distribution are given in Table 2. The relative performances of algorithms were broadly similar to those with a DP mixture model. Algorithm 3 outperformed both Algorithms 4 and 5 with m playing a crucial role in determining the ESS in Algorithm 4. In this case, the Algorithms 4 and 6 have similar ESS's if m is small (in fact, Algorithm 4 with $m = 9$ (results not shown) had a similar ESS to Algorithm 5 for both data sets).

Table 4 The ESS of estimating the posterior mean number of a from 5000 particles with a NGG process prior

Algorithm	Galaxy			Log acidity		
	ESS	Time	R.E.	ESS	Time	R.E.
Algorithm 3	1018	323	3.2	735	581	1.3
Algorithm 4 ($m = 3$)	40	384	0.1	156	745	0.2
Algorithm 4 ($m = 27$)	197	386	0.5	334	754	0.4
Algorithm 4 ($m = 250$)	411	407	1.0	478	755	0.6
Algorithm 5	289	547	0.5	319	1066	0.3

Table 5 The ESS of estimating the posterior mean number of clusters with DP mixture model using different MCMC samplers

Algorithm	Galaxy			Log acidity		
	ESS	Time	R.E.	ESS	Time	R.E.
Pólya urn scheme	801	6	133.5	553	11	50.3
Algorithm 8 ($m = 3$)	686	8	85.8	694	15	46.3
Algorithm 8 ($m = 27$)	545	9	60.6	601	17	35.4
Slice 2	307	11	27.9	234	16	14.6

The previous results assumed a fixed value for the parameter a which effects the modality and shape of the unknown density of the data. Often, we would want to estimate this parameter with the unknown density. Table 3 shows results for the DP mixture model with a given a uniform prior on (0, 1). The ESS was now calculate with the posterior mean of a as the parameter of interest. The ESS's for the non-conjugate methods (Algorithms 4 and 6) were noticeably smaller relative to the ESS for Algorithm 3 compared to the case where a was known. Between the methods for non-conjugate mixtures, Algorithm 4 provided the largest ESS for the two data sets but only when m was large and the Algorithm 5 method provided a much ESS to Algorithm 4 than the case where a was known.

The parameter γ in the NGG process prior controls the flatness of the prior on the number of clusters in a sample of size n (Lijoi et al. 2007). Larger values of γ favouring a larger number of clusters of which many have a small size. The results with an NGG proces prior with $\gamma = 0.2$ on the mixing distribution are shown in Table 4. These indicated a broadly similar pattern of results to those for the DP mixture model but with slightly larger ESS values. These results indicated that all SMC algorithms gave good performance for posterior computation and that a large value of m was preferable for Algorithm 4.

Table 5 shows the ESS (estimated using the initial positive sequence estimator of Geyer 1992) and computational times for three MCMC methods for NRMI mixtures: Conjugate marginalized sampler (Favaro and Teh 2013), the generalization of Neal's algorithm 8 method to NRMI mixture models with m auxiliary variables (Favaro and Teh 2013) and the Slice 2 sampler (Griffin and Walker 2011) for non-conjugate mixture models. The results clearly showed that the MCMC methods dominated the SMC methods in terms of the relative efficiency for both conjugate and non-conjugate DP mixture model samplers. The results for posterior inference about the parameter a in the DP mixture models using MCMC are shown in Table 6. In the conjugate methods, the SMC method had a larger relative efficiencies than the MCMC method for both data sets. The SMC method was 1.9 times more efficient for the log acidity data and 4.5 times more efficient for the galaxy data. However, in non-conjugate methods, all MCMC methods were more efficient than all SMC methods for both data sets. The difference in ordering of relative efficiency performance of SMC and MCMC methods for conjugate and non-conjugate models can be explained by two factors. Firstly, the non-conjugate SMC methods have between two and three time longer computational times than MCMC methods. This difference is not explained by differences in computational complexity and is probably due to implementation issues in Matlab. Secondly, the MCMC methods have similar ESS for conjugate and non-conjugate mixture models but SMC methods have much larger ESS for conjugate than non-conjugate mixture models.

Table 6 The ESS of estimating the posterior mean of a with DP mixture model using different MCMC samplers

Algorithm	Galaxy			Log acidity		
	ESS	Time	R.E.	ESS	Time	R.E.
Pólya urn scheme	75	8	9.4	80	13	6.2
Algorithm 8 ($m = 3$)	35	9	3.9	80	16	5.0
Algorithm 8 ($m = 27$)	64	10	6.4	101	18	5.6
Slice 2	37	12	3.1	61	20	3.1

Particle Gibbs methods were described in Sect. 4. Gibbs sampler with four conditional particle filters were considered with different resampling schemes: multinomial resampling, stratified resampling, adaptive multinomial resampling and adaptive stratified resampling. The methods were run on the infinite mixture model in (12) with a fixed value of a (chosen as in the SMC examples). The results are shown in Fig. 1. Some results with a small number of particles have been excluded due to biased results produced in the runs. Multinomial resampling (Andrieu et al. 2010) led to relatively low ESS's for both data sets. Stratified resampling (Chopin and Singh 2013) led to much larger ESS's with a roughly ten-fold increase in the ESS's compared to multinomial resampling. The addition of an adaptive updating step led to improved ESS's for both resampling methods. The addition of adaptive updating led to a larger improvement for stratified resampling with the log acidity data than the galaxy data. Overall, the difference between the two resampling schemes is small if adaptive updating is included. The methods with adaptive

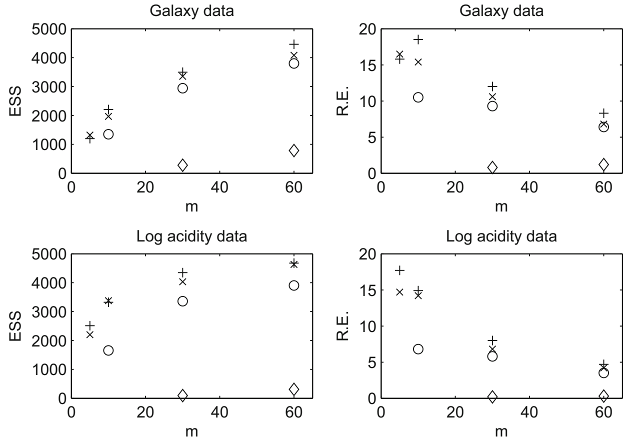


Fig. 1 The ESS and relative efficiencies of estimating the posterior mean number of clusters with DP mixture model particle Gibbs samplers with different re-weighting schemes with m particles. The schemes were: multinomial (*diamond*), stratified (*circle*), adaptive multinomial (*plus*), and adaptive stratified (*times*)

Table 7 The ESS of estimating the posterior mean number of clusters with NGG process prior using different MCMC samplers

Algorithm	Galaxy			Log acidity		
	ESS	Time	R.E.	ESS	Time	R.E.
Pólya urn scheme	924	8	115.5	683	14	48.8
Algorithm 8 ($m = 3$)	880	9	97.8	522	16	32.6
Algorithm 8 ($m = 27$)	822	10	82.2	645	18	35.8
Slice 2	333	52	6.4	369	56	6.6

Table 8 The ESS of estimating the posterior mean of a with an NGG process prior using different MCMC samplers

Algorithm	Galaxy			Log acidity		
	ESS	Time	R.E.	ESS	Time	R.E.
Pólya urn scheme	63	10	6.3	75	14	5.4
Algorithm 8 ($m = 3$)	58	10	5.8	68	16	4.3
Algorithm 8 ($m = 27$)	167	11	15.2	89	28	3.2
Slice 2	59	41	1.4	108	56	1.9

updating led to ESS's in the thousands with only $m = 5$ particles. The ESS is larger than the ESS for the conjugate methods with either SMC or MCMC. However, the computational time is much larger in the current implementation. Interestingly, the ESS's with adaptive resampling and 60 particles were over 4000 indicating that the draws were close to independent (Tables 7, 8).

5.2 Nonparametric stochastic volatility modelling

Stochastic volatility models are a popular approach to modelling a time series of prices of a financial asset, $p_1, \ldots, p_T$ recorded over a fixed period time (e.g. daily). In a simple stochastic volatility model, the log returns $r_t = \log p_{t+1} - \log p_t$ are modelled as

$$r_t = \beta \exp\{h_t/2\}\epsilon_t$$

and

$$h_{t+1} = \mu + \phi(h_t - \mu) + \nu_t$$

where $(\epsilon_t, \nu_t) \overset{ind.}{\sim} F$. The variance of r_t conditional on h_t is $\beta^2\psi \exp\{h_t\}$ where $\psi = \mathrm{V}[\epsilon_t]$ and so h_t, which is called the log volatility, allows the conditional variance of r_t to change over time. The model is usually made identifiable by setting $\beta = 1$ or $\mu = 0$. The distribution G is often assumed to be a bivariate normal distribution. A non-zero correlation between ϵ_t and ν_t allows modelling of the leverage effect, which is the empirically observed difference in the effect on log volatility of negative and positive log returns of the same magnitude. Bayesian nonparametric approaches to estimating the distribution of ϵ_t are described in Jensen and Maheu (2010) and Delatola and Griffin (2011) and to the estimation of the joint distribution of ϵ_t and ν_t are described by Jensen and Maheu (2014) and Delatola and Griffin (2013). I consider a slight variation on the model of Jensen and Maheu (2014)

$$\begin{aligned}&\begin{pmatrix}\epsilon_t\\ \nu_t\end{pmatrix}\Bigg|\, \mu_{t,1:2}, \rho_t\\ &\quad\sim \mathrm{N}\left(\begin{pmatrix}\mu_{t,1}\\ \mu_{t,2}\end{pmatrix}, \begin{pmatrix} a_1\sigma^2 & \sqrt{a_1a_2}\sigma\sigma_h\rho_t\\ \sqrt{a_1a_2}\sigma\sigma_h\rho_t & a_2\sigma_h^2\end{pmatrix}\right),\\ &\quad(\mu_{t,1:2}, \rho_t) \sim G\end{aligned}$$

where G is a given a DP prior with $M = 1$ and centring measure

$$\begin{aligned}&\mathrm{N}\left(\begin{pmatrix}\mu_1\\ \mu_2\end{pmatrix}\Bigg|\begin{pmatrix}0\\ 0\end{pmatrix}, \begin{pmatrix}(1-a_1)\sigma^2 & \left(1-\sqrt{a_1a_2}\right)\sigma\sigma_h\rho\\ \left(1-\sqrt{a_1a_2}\right)\sigma\sigma_h\rho & (1-a_2)\sigma_h^2\end{pmatrix}\right)\\ &\quad\times \mathrm{TN}_{-1,1}(\rho|\mu_\rho, \sigma_\rho^2),\end{aligned}$$

$\mathrm{TN}_{a,b}(\mu, \sigma^2)$ represents a normal distribution with mean μ and variance σ^2 truncated to (a, b) and $0 < a_1 < 1$ and $0 < a_2 < 1$. The model allows different values of μ_1, μ_2 and ρ (which is the correlation) in each component and so allows for a non-normal joint distribution of η_t and ν_t and a non-linear leverage effect. The priors are $\phi \sim \mathrm{Be}(20, 1.5)$, $\sigma_h^{-2} \sim \mathrm{Ga}(0.1, 0.1)$, $\sigma^{-2} \sim \mathrm{Ga}(0.1, 0.1)$, $\mu_\rho \sim \mathrm{U}(-1, 1)$ and $\sigma_\rho^2 \sim \mathrm{Ga}(1, 100)$. This implies that the prior mean of σ_ρ^2 is 0.01 and supports small differences in the correlation between different components.

The model is non-conjugate and an extension to Algorithm 5 to allow parameter updating was used to fit data from the FTSE 100 index from 17 May 2008 to 1 May 2012, which had

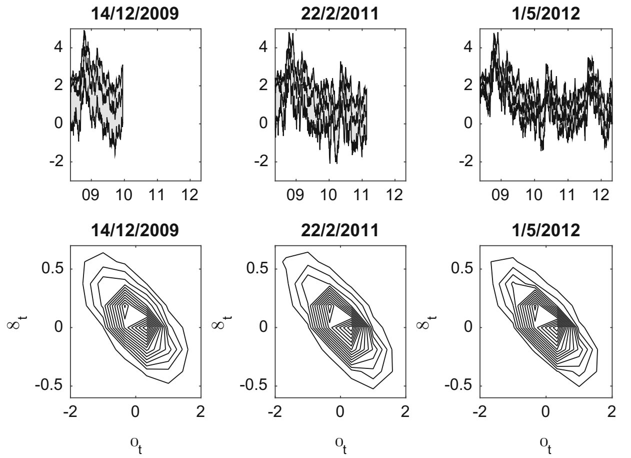


Fig. 2 Estimated return distribution and log volatility at three dates: 14/12/2009, 22/2/2011 and 1/5/2012. The *top row* shows the filtered median log volatility (*solid line*) with 95 % credible interval estimated at each date and the *bottom row* shows the filtered mean return distribution at each date

1000 observations. We use $a_1 = a_2 = 0.1$ which allows quite substantial departures from bivariate normality (see Griffin 2010, for more details in the univariate context) and 5000 particles. The posterior distribution of the log volatility and the posterior mean joint density of η_t and ν_t are shown at three dates: 14 December 2009, 22 February 2011 and 1 May 2012 which are the 400th, 700th and 1000th (final) returns. The posterior mean densities show clear dependence and so accommodate the leverage effect. The results also show a much stronger negative dependence for more extreme values of ϵ_t. However, the estimates seem to be very similar at the three different time points.

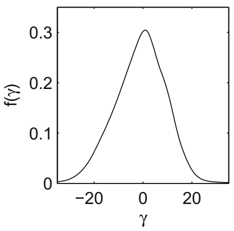
5.3 Testing a parametric model against a nonparametric alternative

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 DP based (rather than mixture of DP 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 (Fig. 2).

The "schoolgirls" data set of the DPpackage in R records the heights of 20 girls at ages 6–10 (in years). We consider the problem of specifying a random effects model which has the form

$$\begin{aligned} y_{i,t} &= \beta_0 + (t - \bar{t})\beta_1 + \gamma_i + \epsilon_{i,t}, \\ i &= 1, \ldots, n, \quad t = 1, \ldots, T \end{aligned}$$

Fig. 3 The posterior mean distribution of the random effects in the nonparametric model applied to the "schoolgirl" data set



where t is the age, $\bar{t} = 8$ is the average age, $\gamma_i \sim F$ is a random effect and $\epsilon_{i,t} \sim \mathrm{N}(0, \sigma^2)$. A parametric specification where F is a normal distribution with mean zero and variance σ^2_γ is tested against a nonparametric alternative where

$$\gamma_i \sim \mathrm{N}\left(\mu_i, a\sigma^2_\gamma\right), \quad \mu_i \sim G, \quad G \sim \mathrm{DP}(MH)$$

and H is a normal distribution with mean 0 and variance $(1 - a)\sigma^2_\gamma$. The parameter a is set equal to 0.03 to allow for a wide-range of distributions of the random effects. The other priors are common to both models: $\beta = (\beta_0, \beta_1)^T \sim \mathrm{N}(0, 100^2)$, $\sigma^{-2} \sim \mathrm{Ga}(0.01, 0.01)$ and $\sigma_\gamma^{-2} \sim \mathrm{Ga}(0.01, 0.01)$. The posterior mean of F for the nonparametric model is shown in Fig. 3 and indicates a departure from a normal distribution. To test the strength of this effect, we run Algorithm 3 to calculate the log marginal likelihood for the nonparametric model which is estimated to be −219.8. The log marginal likelihood for the parametric model can be estimated using a SMC giving a value of −218.5. This implies that the Bayes factor in favour of the parametric model is $e^{1.3} = 3.7$ which represents weak evidence against the nonparametric model.

6 Discussion

There has been little work on the use of SMC methods for fitting nonparametric mixture models which are not based on Dirichlet processes. This paper has described SMC methods for the wide-class of NRMI mixture models with both conjugate and non-conjugate structure. These can be used to estimate nonparametric mixture models sequentially, estimate marginal likelihoods or as components in particle Gibbs samplers. The results suggest that SMC methods work well in conjugate mixture models. In particular, SMC methods can outperform Gibbs samplers for parameter estimation problems in static inference problems. I have considered two methods for non-conjugate mixture models: one based on slice sampling (Algorithms 5 and 6) and one based on marginalization (Algorithm 4). The marginalization method tends to outperform the slice sampling methods. Both methods provide useful inference in the problems considered. The number of auxiliary variables (m) plays an important role in the sampler. Suitable choice will depend on the problem at hand but large values of m (in the hundreds) seem appropriate if values from $H_t^\star$ can be generated cheaply. Particle Gibbs methods are an interesting approach for mixture models since these can jointly update $s_1^\star, \ldots, s_n^\star$ in marginalized samplers. The results in this paper indicate that the resampling mechanism can have a substantial effect on the performance of the algorithm. Adaptive resampling methods perform best in the examples considered in this paper. These can produce relatively uncorrelated samples with small numbers of particles (an ESS over 1000 with five particles in both examples) and near independent samples with a relatively small number of particles. This is encouraging and is a promising direction for future research. The NRMI class of priors underlies recently developed time-series and spatial nonparametric priors (see e.g. Griffin et al. 2013; Chen et al. 2013; Lijoi et al. 2014; Bassetti et al. 2014) and extensions of SMC methods to these models will be an area of future research.

References

Andrieu, C., Doucet, A., Holenstein, R.: Particle Markov chain Monte Carlo methods (with discussion). J. R. Stat. Soc. Ser. B **72**(3), 269–342 (2010)

Bassetti, F., Casarin, R., Leisen, F.: Beta-product dependent Pitman–Yor processes for Bayesian inference. J. Econom. **180**, 49–72 (2014)

Basu, S., Chib, S.: Marginal likelihood and Bayes factors for Dirichlet process mixture models. J. Am. Stat. Assoc. **98**, 224–235 (2003)

Berger, J., Guglielmi, A.: Testing of a parametric model versus nonparametric alternatives. J. Am. Stat. Assoc. **96**, 174–184 (2001)

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

Brix, A.: Generalized gamma measures and shot-noise Cox processes. Adv. Appl. Probab. **31**, 929–953 (1999). ISSN 0001-8678

Caron, F., Davy, M., Doucet, A., Duflos, E., Vanheeghe, P.: Bayesian inference for linear dynamics models with Dirichlet process mixtures. IEEE Trans. Signal Process. **56**, 71–84 (2008)

Carota, C., Parmigiani, G.: On Bayes factors for nonparametric alternatives. In: Bernardo, J.M., Berger, J.O., Dawid, A.P., Smith, A.F.M. (eds.) Bayesian Statistics 5, pp. 508–511. Oxford University Press, London (1996)

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

Carvalho, C.M., Lopes, H.F., Polson, N.G., Taddy, M.A.: Particle learning for general mixtures. Bayesian Anal. **5**, 709–740 (2010)

Chen, C., Rao, V.A., Buntine, W., Teh, Y.W.: Dependent normalized random measures. In: Proceedings of the International Conference on Machine Learning (2013)

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

Chopin, N., Singh, S.S.: On the particle Gibbs sampler. arXiv:1304.1887v1 (2013)

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

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

Del Moral, P., Doucet, A., Jasra, A.: Sequential Monte Carlo samplers. J. R. Stat. Soc. Ser. B **68**, 411–436 (2006)

Delatola, E.-I., Griffin, J.E.: Bayesian nonparametric modelling of the return distribution with stochastic volatility. Bayesian Anal. **6**, 901–926 (2011)

Delatola, E.-I., Griffin, J.E.: A Bayesian semiparametric model for volatility modelling with a leverage effect. Comput. Stat. Data Anal. **60**, 97–110 (2013)

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

Favaro, S., Teh, Y.W.: MCMC for normalized random measure mixture models. Stat. Sci. **28**, 335–359 (2013)

Fearnhead, P.: Particle filters for mixture models with an unknown number of components. Stat. Comput. **14**, 11–21 (2004)

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

Geisser, S., Eddy, W.F.: A predictive approach to model selection. J. Am. Stat. Assoc. **74**, 153–160 (1979)

Geyer, C.: Practical Markov chain Monte Carlo. Stat. Sci. **7**, 473–511 (1992)

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

Griffin, J.E., Walker, S.G.: Posterior simulation of normalised random measure mixtures. J. Comput. Graph. Stat. **20**, 241–259 (2011)

Griffin, J.E., Kolossiatis, M., Steel, M.F.J.: Comparing distributions by using dependent normalized random-measure mixtures. J. R. Stat. Soc. Ser. B **75**, 499–529 (2013)

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

James, L., Lijoi, A., Prünster, I.: Posterior analysis for normalized random measures with independent increments. Scand. J. Stat. **36**, 76–97 (2009)

Jensen, M.J., Maheu, J.M.: Bayesian semiparametric stochastic volatility modeling. J. Econom. **157**, 305–316 (2010)

Jensen, M.J., Maheu, J.M.: Estimating a semiparametric asymmetric stochastic volatility model with a Dirichlet process mixture. J. Econom. **178**, 523–538 (2014)

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

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

Lijoi, A., Mena, R.H., Prünster, I.: Controlling the reinforcement in Bayesian non-parametric mixture models. J. R. Stat. Soc. Ser. B **69**, 715–740 (2007). ISSN 1369-7412

Lijoi, A., Nipoti, B., Prünster, I.: Bayesian inference with dependent normalized completely random measures. Bernoulli **20**, 1260–1291 (2014)

Lindsten, F., Jordan, M.I., Schön, T.B.: Particle Gibbs with ancestor sampling. J. Mach. Learn. Res. **15**, 2145–2184 (2014)

Liu, J.S.: Nonparametric hierarchical Bayes via sequential imputation. Ann. Stat. **24**, 910–930 (1996)

MacEachern, S.N., Müller, P.: Estimating mixture of Dirichlet process models. J. Comput. Graph. Stat. **7**, 223–238 (1998)

MacEachern, S.N., Clyde, M.A., Liu, J.: Sequential importance sampling for nonparametric Bayes models: the next generation. Can. J. Stat. **27**, 251–267 (1999)

McVinish, R., Rousseau, J., Mengersen, K.: Bayesian goodness of fit testing with mixtures of triangular distributions. Scand. J. Stat. **36**, 337–354 (2009)

Neal, R.: Markov chain sampling methods for Dirichlet process mixture models. J. Comput. Graph. Stat. **9**, 249–265 (2000)

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

Pitt, M.K., Shephard, N.: Filtering via simulation: auxiliary particle filters. J. Am. Stat. Assoc. **94**, 590–599 (1999)

Ulker, Y., Gunsel, B., Taylan Cemgil, A.: Sequential Monte Carlo samplers for Dirichlet process mixtures. J. Mach. Learn. Res. **9**, 876–883 (2010)

Walker, S.G.: Sampling the Dirichlet mixture model with slices. Commun. Stat. Simul. Comput. **36**(1–3), 45–54 (2007)

Whiteley, N.: Discussion of "Particle Markov chain Monte Carlo methods". J. R. Stat. Soc. Ser. B **72**(3), 306–307 (2010)

Whiteley, N., Andrieu, C., Doucet, A.: Efficient Bayesian inference for switching state-space models using discrete particle Markov chain Monte Carlo methods. Technical Report, University of Bristol (2010)