

Sequential Monte Carlo methods and perfect sampling for mixture models

Paul Fearnhead

Department of Mathematics and Statistics,
Lancaster University

(Joint work with Loukia Meligkotsidou)
www.maths.lancs.ac.uk/~fearnhea

Talk Outline

- Exact Simulation for Discrete Data
- Sequential Monte Carlo methods

Based on work in [Fearnhead \(2005\)](#) and [Fearnhead \(2004\)](#) respectively.

Related recent work on SMC by [Taddy, Carvalho, Lopes and Polson](#).

Mixture Models

We consider Bayesian inference for mixture models of the form

$$p(y|\theta, \mathbf{p}) = \sum_{i=1}^K p_i f(y|\theta_i),$$

from data $\mathbf{y} = (y_1, y_2, \dots, y_N)$.

We assume conjugate priors for \mathbf{p} (Dirichlet) and θ , denoted by $\pi(\theta, \mathbf{p})$.

We will introduce latent variable \mathbf{z} , such that z_i denotes the component of observation y_i ; so

$$p(y_i|\theta, \mathbf{p}, z_i = k) = f(y_i|\theta_k)$$

.

Posterior Distribution

We can write the posterior as a **sum over the possible values** of \mathbf{z} . For a given \mathbf{z} let $\mathbf{n}(\mathbf{z})$ be a vector of the number of observations allocated to each component; and $\mathbf{s}(\mathbf{z})$ the vector of sufficient statistics for each component.

Then

$$\begin{aligned} p(\theta, \mathbf{p}|\mathbf{y}) &\propto \sum_{\mathbf{z}} \pi(\theta, \mathbf{p}) p(\mathbf{z}|\mathbf{p}) p(\mathbf{y}|\theta, \mathbf{z}) \\ &\propto \sum_{\mathbf{z}} g(\theta, \mathbf{p}, \mathbf{n}(\mathbf{z}), \mathbf{s}(\mathbf{z})), \end{aligned}$$

for some function g .

Note that $\pi(\theta, \mathbf{p}) p(\mathbf{z}|\mathbf{p}) p(\mathbf{y}|\theta, \mathbf{z}) \propto p(\mathbf{z}|\mathbf{y}) p(\theta, \mathbf{p}|\mathbf{y}, \mathbf{z})$

Example: Poisson mixtures

Consider a **Poisson mixture**. Assume **independent Gamma priors** on the component means. Let α be the vector of parameters of the Dirichlet prior, and let β_k and γ_k be the parameters of the Gamma prior for θ_k , the mean of the k th component.

The **sufficient statistics** $\mathbf{s}(\mathbf{z})$ are the **sum of observations** allocated to each component.

The posterior distribution can be written as

$$p(\mathbf{z}|\mathbf{y}) \propto \prod_{k=1}^K \left(\Gamma(\alpha_k + n_k) \Gamma(\beta_k + s_k) (\gamma_k + n_k)^{-(\beta_k + s_k)} \right),$$
$$p(\theta, \mathbf{p}|\mathbf{y}, \mathbf{z}) = \text{Dir}(\mathbf{p}; \alpha + \mathbf{n}) \prod_{k=1}^K \text{Gam}(\theta_k; \beta_k + s_k, \gamma_k + n_k),$$

Where we have written $\mathbf{n} = (n_1, \dots, n_K)$ for $\mathbf{n}(\mathbf{z})$, and similarly for \mathbf{s} .

Example: Poisson mixtures

And we have

$$g(\boldsymbol{\theta}, \mathbf{p}, \mathbf{n}(\mathbf{z}), \mathbf{s}(\mathbf{z})) = \text{Dir}(\mathbf{p}; \boldsymbol{\alpha} + \mathbf{n}) \prod_{k=1}^K \left(\frac{\Gamma(\alpha_k + n_k) \Gamma(\beta_k + s_k)}{(\gamma_k + n_k)^{(\beta_k + s_k)}} \text{Gam}(\theta_k; \beta_k + s_k, \gamma_k + n_k) \right).$$

Where $\text{Dir}(\mathbf{x}; \boldsymbol{\alpha})$ denotes the pdf of a Dirichlet distribution with parameters $\boldsymbol{\alpha}$ evaluated at \mathbf{x} ; and similarly Gam denote the pdf of a Gamma distribution.

The key idea is that for discrete data (as here) many allocations \mathbf{z} may produce the same summaries $\mathbf{n}(\mathbf{z}), \mathbf{s}(\mathbf{z})$.

Reformulation of Posterior

Our idea is that if we can calculate the number of allocations \mathbf{z} that produce summaries \mathbf{n}, \mathbf{s} , which we denote $M_N(\mathbf{n}, \mathbf{s})$, then

$$p(\theta, \mathbf{p} | \mathbf{y}) \propto \sum_{\mathbf{n}, \mathbf{s}} M_N(\mathbf{n}, \mathbf{s}) g(\theta, \mathbf{p}, \mathbf{n}, \mathbf{s}).$$

It is possible to calculate the $M_N(\mathbf{n}, \mathbf{s})$ through recursions that calculate $M_i(\mathbf{n}, \mathbf{s})$ from $M_{i-1}(\mathbf{n}, \mathbf{s})$ for $i = 1, \dots, N$.

This gives a feasible approach to calculating the posterior providing the number of distinct values of \mathbf{n}, \mathbf{s} is not too large.

Example: Fetal Lamb Movements

We applied our method to data on the number of movements of a fetal lamb over 240 consecutive 5 second periods (data from [Leroux and Puterman 1992](#)).

The data is:

Movements	0	1	2	3	4	5	6	7
Frequency	182	41	12	2	2	0	0	1

We fit [1, 2 and 3 component Poisson mixture models](#), and a 2 component model which allowed for [Markov dependence](#) in the allocations over time.

Results

K	Log Evidence	Poisson means	Mixture Proportion
1	-203.8	0.36 (0.0015)	1 (0)
2	-192.7	0.18 (0.0042) 1.72 (0.43)	0.86 (0.0088) 0.14 (0.0088)
2(M)	-185.7	0.23 (0.0027) 1.89 (1.3)	0.92 (0.020) 0.80 (0.024)
3	-192.0	0.12 (0.0049) 0.49 (0.11) 2.02 (0.63)	0.55 (0.060) 0.35 (0.059) 0.10 (0.048)

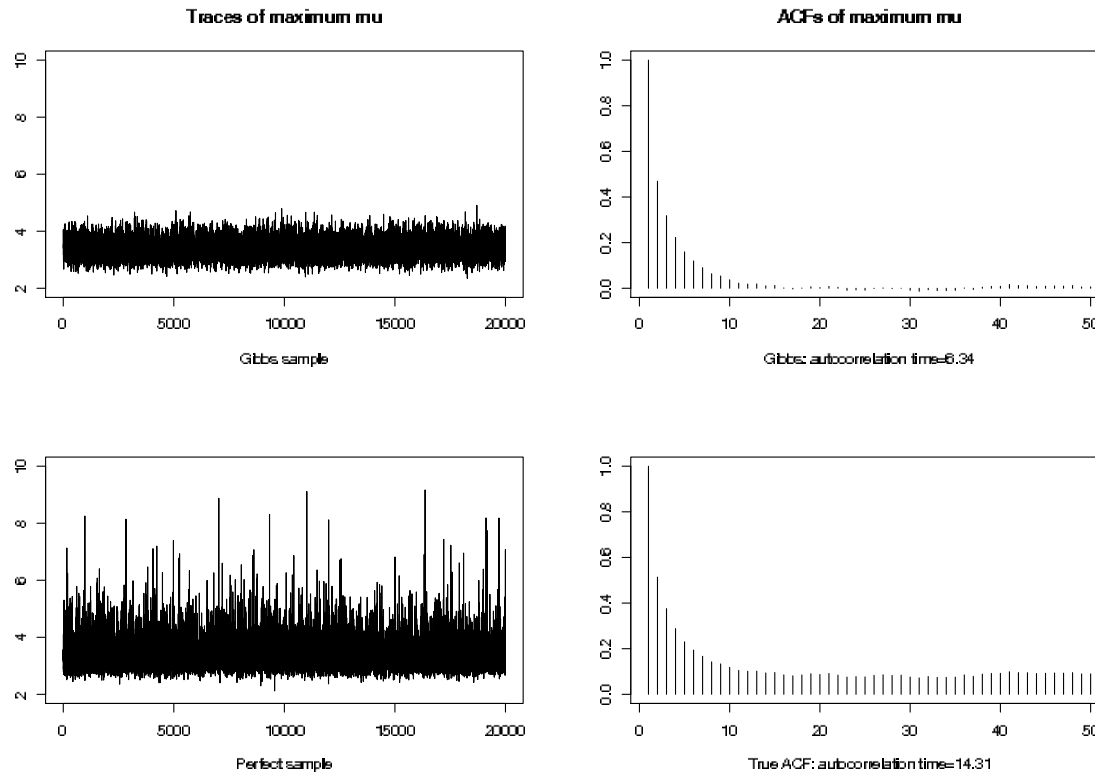
Computational Feasibility

Analysis took from seconds ([2 component mixture](#)) to an hour ([Markov model and 3 component mixture](#)).

The posterior for the 3 component mixture has [12 million](#) terms in; and the algorithm uses [1.3Gb](#).

Range of application can be extended by grouping allocations with similar summaries – including to analysing continuous mixtures. Approximations can be corrected by Importance Sampling or Rejection Sampling. (See [Fearnhead and Meligkotsidou 2007](#).)

Does it Matter?



Analysis of 2 component Gaussian mixture by MCMC and Exact Simulation.

Sequential Monte Carlo Methods for Mixture Models

We now consider sequential analysis of data. Let $\mathbf{z}_{1:i} = (z_1, z_2, \dots, z_i)$. As before we assume *conjugate* priors so that we can analytically calculate $p(\theta, \mathbf{p} | \mathbf{z}_{1:i}, \mathbf{y}_{1:i})$, and $p(y_{1:i} | \mathbf{z}_{1:i})$ for all i .

SMC can be applied both to inference for models with a fixed number of components K , and to *Dirichlet Process Mixture* models.

Sequential Monte Carlo

Our approach is to use Sequential Monte Carlo to generate approximations to $p(\mathbf{z}_{1:i}|\mathbf{y}_{1:i})$ for $i = 1, 2, \dots, N$.

Given a set of weighted particles $\mathbf{z}_{1:i}^{(j)}, w_i^{(j)}$, for $j = 1, \dots, M$; we approximate $p(\mathbf{z}_{1:i}|\mathbf{y}_{1:i})$ by the distribution which assigns $\mathbf{z}_{1:i}^{(j)}$ a probability $w_i^{(j)}$.

Sequential Monte Carlo gives a method for generating the weighted particles at time $i + 1$ given those at time i .

Filtering Recursions

For any particle $\mathbf{z}_{1:i}^{(j)}$, we can calculate the optimal proposal $p(z_{i+1} | \mathbf{y}_{1:i+1}, \mathbf{z}_{1:i}^{(j)})$.

We can then simulate $z_{i+1}^{(j)}$ from this distribution to obtain a new particle

$$\mathbf{z}_{1:i+1}^{(j)} = (\mathbf{z}_{1:i}^{(j)}, z_{i+1}^{(j)}),$$

and a new weight

$$w_{i+1}^{(j)} = w_i^{(j)} p(y_{i+1} | \mathbf{z}_{1:i}^{(j)}, \mathbf{y}_{1:i})$$

.

This together with a resampling step is the algorithm of [Chen and Liu \(2000\)](#).

Improved Method

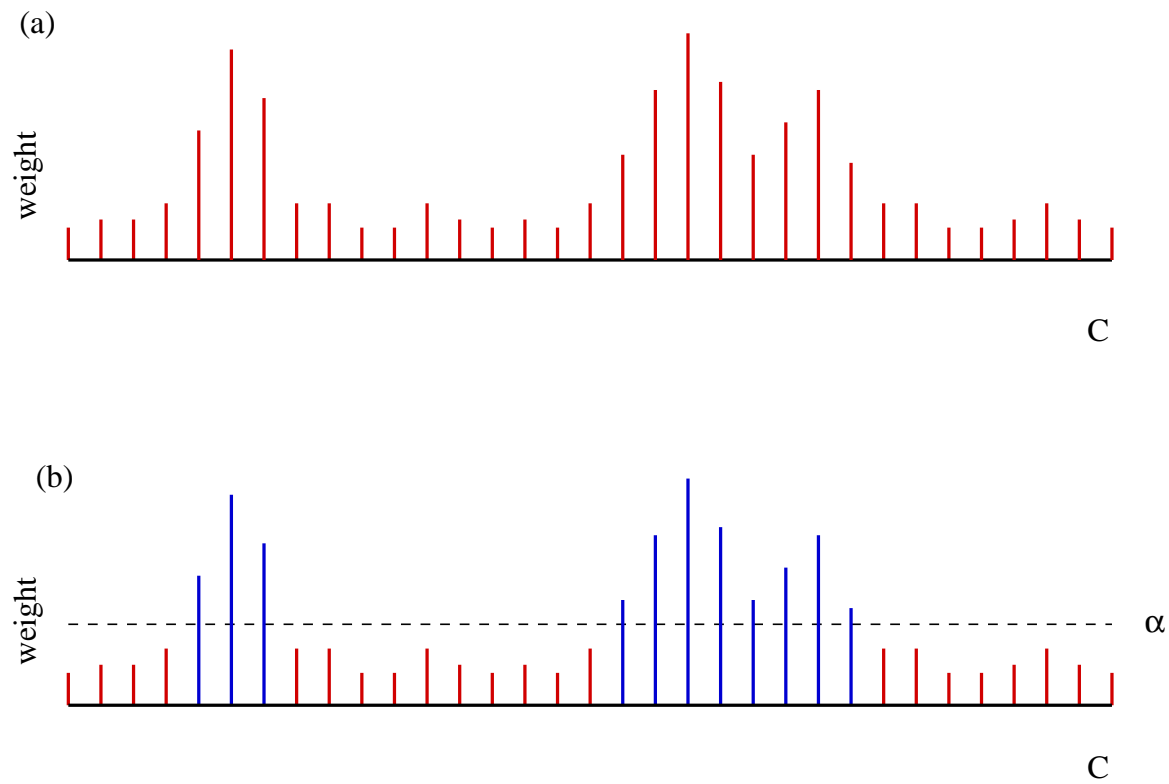
However to calculate $p(z_{i+1} | \mathbf{y}_{1:i+1}, \mathbf{z}_{1:i}^{(j)})$ requires calculating $p(\mathbf{y}_{1:i+1} | z_{i+1} = k, \mathbf{z}_{1:i}^{(j)})$ for all k .

So for the same CPU cost we can calculate all possible particles, and their weights at time $i + 1$ that are consistent with the particles at time i .

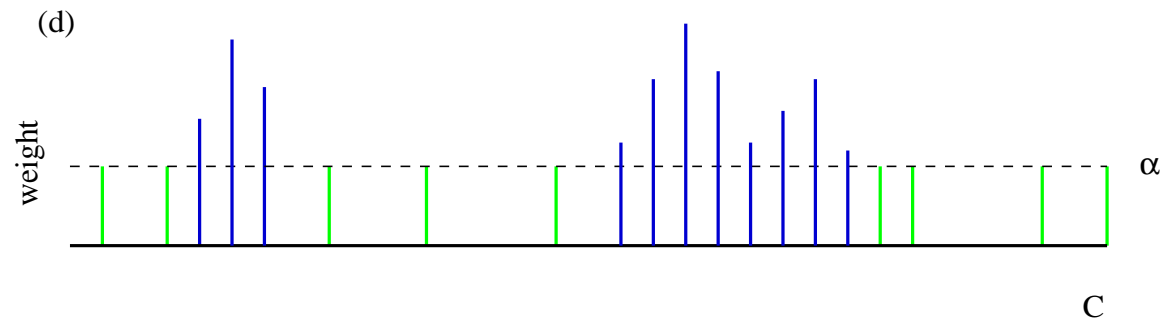
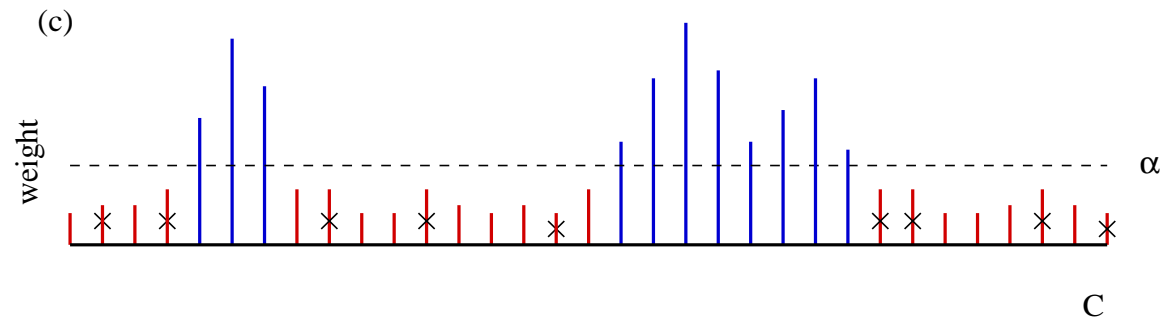
To avoid the number of particles increasing by a factor of K each time, we then use resampling to remove particles, and reweight those that we keep.

Resampling Algorithm

An *optimal* way to resample is given by [Fearnhead and Clifford \(2003\)](#).



Resampling Algorithm



Example

We simulated data from a 3-component Gaussian mixture model:

$$X_t \sim \begin{cases} N(0, 0.5^2) & \text{with probability } 1/2, \\ N(\mu, 0.5^2) & \text{with probability } 1/6, \\ N(2\mu, \sigma^2) & \text{with probability } 1/3. \end{cases}$$

We analysed 8 data sets corresponding to different values of μ and σ . Our model was a Dirichlet Process Mixture model (unknown K).

For each data set we compared our SMC method, with that of Chen and Liu (2000), and with a Gibbs Sampler. Comparisons were based on an Effective Sample Size, calculated from the variability of each method across independent runs.

Results

	$\sigma = 0.5$				$\sigma = 2.5$			
	$\mu = 0.5$	$\mu = 1$	$\mu = 2$	$\mu = 5$	$\mu = 0.5$	$\mu = 1$	$\mu = 2$	$\mu = 5$
Our SMC	1.8×10^5	1,200	160	3.0×10^5	850	41	3,900	1,900
Chen and Liu	5,500	320	18	750	350	13	1,700	140
Gibbs Sampler	590	580	350	720	780	410	720	790

Results for 5,000 particles, or 5,500 iterations of the Gibbs sampler. The Gibbs sampler also works on the space of allocations, and updates z_i given \mathbf{z}_{-i} .

Discussion

It is possible to combine the ideas of exact simulation with the SMC approach ([Fearnhead and Meligkotsidou 2007](#)).

For discrete mixtures we can have particles that store $(\mathbf{n}(\mathbf{z}), \mathbf{s}(\mathbf{z}))$, rather than \mathbf{z} – this reduces the rate at which the number of particles increases at each iteration before resampling; and hence the approximation introduced by resampling.

For continuous mixtures, this is similar to implementing [stratified resampling](#), which takes account of the value of $(\mathbf{n}(\mathbf{z}), \mathbf{s}(\mathbf{z}))$.

References

- Casella, G., Mengerson, K., Robert, C.P. and Titterington, D.M. (2002) Perfect samplers for mixture distributions. *JRSS B* **64** p.777.
- Chen, R. and Liu, J. (2000) Mixture Kalman Filters. *JRSS B* **62**, p.493
- Fearnhead, P. (2004) Particle filters for mixture models with an unknown number of components. *Stat. Comp.* **14**, p.11
- Fearnhead, P. (2005) Direct simulation for discrete mixture distributions. *Stat. Comp.* **15**, p.125
- Fearnhead, P. and Clifford, P. (2003) Online inference for hidden Markov models. *JRSS B* **65**, p.887
- Fearnhead, P. and Meligkotsidou, L. (2007) Filtering methods for Mixture Models. *JCGS* **16**, p.586
- Hobert, J., Robert, C.P. and Titterington, D.M. (1999) On perfect simulations for some mixtures of distributions. *Stat. Comp.* **9**, p.287
- Leroux, B.G. and Puterman, M.L. (1992) Maximum penalized-likelihood estimation for independent and Markov dependent mixture models. *Biometrics* **51**, p.1033.