Noname manuscript No.
(will be inserted by the editor)

MCMC estimation of a finite beta mixture

Andriy Norets · **Xun Tang**

October 27, 2010

Abstract We describe an efficient Markov chain Monte Carlo algorithm for estimation of a finite beta mixture. The algorithm employs Metropolis-Hastings independence chain for simulation of the parameters of beta distributions. The Metropolis-Hastings transition densities that well approximate the target distributions are constructed from the limiting sampling distribution of the method of moments estimator, which is readily available for beta distribution. This technique can be useful for other models with analytically tractable method of moments estimators. The algorithm demonstrated excellent performance in a Monte-Carlo study.

Keywords MCMC · finite mixture · beta distribution · method of moments

1 Introduction

Finite mixtures of distributions are widely used as flexible models for univariate and multivariate data (McLachlan and Peel (2000)). It is well known that a finite beta mixture can consistently estimate densities on [0,1] from large nonparametric classes, see Rousseau (2010) for recent rates of convergence results. Bouguila et al. (2006) described a Markov chain Monte Carlo (MCMC) algorithm for estimation of a finite beta mixture.[1] Their algorithm follows the Diebolt and Robert (1994) approach of using latent mixture component indicators and data augmentation in estimation of finite mixture models. In Bouguila et al. (2006) algorithm, the parameters of beta distributions are simulated by a Metropolis-Hastings random walk algorithm. Below, we describe a more efficient Metropolis-Hastings independence chain algorithm for simulation of the parameters of beta distributions. Our algorithm employs Metropolis-Hastings transition density based on the sampling distribution of the method of moments (MOM) estimator for the parameters of beta.

The Bernstein-von Mises theorem suggests that the posterior distribution can be approximated by a normal distribution with a mean equal to the maximum likelihood estimator (MLE) and a variance equal to the sampling variance of the MLE. The issue of model misspecification, which would require a modification of the Bernstein-von Mises theorem, does not seem to be important here as a finite beta mixture is a very flexible model. Unfortunately, the MLE is not available in closed form

A. Norets
Department of Economics, Princeton University
E-mail: anorets@princeton.edu

X. Tang
Department of Economics, University of Pensylvania
E-mail: xuntang@sas.upenn.edu

[1] Bouguila et al. (2006) also describe an expectation maximization algorithm for estimation of a beta mixture model.

for the parameters of beta distribution. The MOM estimator and an approximation to its sampling distribution are available analytically. The asymptotic sampling variance of the MOM estimator is in general larger than the variance of the MLE. Thus, the Metropolis-Hastings transition densities we construct are likely to have heavier tails than the target distribution. This is a useful property for an independence chain MCMC algorithm as it implies uniform ergodicity and thus central limit theorems for the MCMC algorithm, see Tierney (1994). The quality of the approximations to the posteriors and thus the performance of the independence chain MCMC algorithm depend on how efficient the MOM estimator is. Experiments demonstrate that the quality of approximations to the conditional posteriors of the parameters of beta distributions in a finite beta mixture model is excellent even for small sample sizes.

More generally, these ideas might be useful for developing MCMC samplers for models in which the MLE is not analytically tractable while the method of moments estimator is. Below, we describe the model and the MCMC algorithm. Applications of the algorithm in the context of a larger hierarchical model can be found in Norets and Tang (2010). The last section presents a Monte Carlo study that compares the performance of the random walk and MOM-based algorithms.

2 The likelihood, prior, and posterior

A random variable $p_i \in (0,1)$ follows a Beta distribution with parameters $(m_j s_j, (1-m_j)s_j)$ if its density is given by

$$f(p_i|s_j,m_j) = \frac{\Gamma(s_j) p_i^{s_j m_j - 1}(1-p_i)^{s_j(1-m_j)-1}}{\Gamma(s_j m_j)\Gamma(s_j(1-m_j))}. \tag{1}$$

It is convenient for our purposes to parameterize a beta distribution in terms of s_j and m_j. A density of a finite beta mixture with M components is defined as

$$\pi(p_i|s,m,\lambda) = \sum_{j=1}^{M} \lambda_j f(p_i|s_j,m_j),$$

where λ_j is the probability that p_i is generated by component j and $s = (s_1, s_2, \ldots, s_M)$ and likewise for m and λ.

Let $p = \{p_1, \ldots, p_N\}$, $p_i \in (0,1)$, denote a vector of observations. The likelihood function for a finite beta mixture is given by

$$\pi(p|m,s,\lambda) = \prod_{i=1}^{N} \sum_{j=1}^{M} \lambda_j f(p_i|s_j,m_j) \tag{2}$$

Let Z_i be a latent variable such that $Z_i = j$ if p_i is drawn from $Beta(s_j, m_j)$. Let $Z = \{Z_i\}_{i \le N}$. The distribution of observables conditional on the parameters and latent variables has a more tractable form than (2),

$$\pi(p|m,s,\lambda,Z) = \prod_i f(p_i|s_{Z_i}, m_{Z_i}). \tag{3}$$

We specify the joint prior of (m,s,λ,Z) as follows: λ is independent from the (s,m) and all coordinates in (s,m) are mutually independent. Furthermore, $m_j \sim Beta(\underline{n}_{m_1}, \underline{n}_{m_0})$, $s_j \sim Gamma(\underline{a}_s, \underline{b}_s)$ for all j and $\lambda \sim Dirichlet(\underline{a}, ., \underline{a})$, where $\underline{n}_{m_1}$, $\underline{n}_{m_0}$, $\underline{a}_s$, $\underline{b}_s$, and $\underline{a}$ are all known positive scalars. The joint prior for (m,s,λ,Z) is then given by

$$\begin{aligned} &\pi(m,s,\lambda,Z) = \pi(Z|m,s,\lambda)\pi(m,s,\lambda) \\ &\propto \prod_j \big[\lambda_j^{\sum_i 1\{Z_i=j\}} \cdot \lambda_j^{\underline{a}-1} \\ &\cdot m_j^{\underline{n}_{m_1}-1}(1-m_j)^{\underline{n}_{m_0}-1} \\ &\cdot s_j^{\underline{a}_s-1} \exp\{-s_j/\underline{b}_s\}\big] \end{aligned} \tag{4}$$

The joint posterior, $\pi(m,s,\lambda,Z|p)$, is proportional to the product of (3) and (4).

3 Posterior Simulations Using MCMC

In this section we describe a Metroplis-within-Gibbs MCMC algorithm for exploring the joint posterior $\pi(m,s,\lambda,Z|p)$.[2] The algorithm divides the vector of parameters and latent variables into the following Gibbs sampler blocks: $\{s_j\}_{j \le M}$, $\{m_j\}_{j \le M}$, $\{Z_i\}_{i \le N}$, and λ. The density (or probability mass function in case of Z_i) for each block is proportional to the product of (3) and (4). The blocks for Z_i and λ

[2] See Tierney (1994) or Geweke (2005) for a discussion of hybrid MCMC algorithms.

are standard (multinomial and Dirichlet distributions correspondingly). The distributions of blocks for s_j and m_j do not seem to have known closed forms. Therefore, we use a Metropolis-Hastings algorithm for these blocks. If good approximations to the conditional posteriors of s_j and m_j are available one can construct an efficient Metropolis-Hastings independence chain algorithm, in which the approximations to the conditional posteriors serve as the Metropolis-Hastings transition densities. In the introduction we explain why the sampling distribution of the method of moment estimator provides a good approximation to the posterior distribution. The implied approximations to the conditional posteriors of s_j and m_j are normal (we derive them in Appendix A below). Since the supports of s_j and m_j ($[0,\infty)$ and $[0,1]$ correspondingly) are not the same as the support of a normal, we use a beta transition density for m_j and a gamma transition density for s_j that have the same means and variances as the corresponding normal approximations. We also take into account the part of the posterior that corresponds to a $Beta(\underline{n}_{m_1},\underline{n}_{m_0})$ prior for m_j and a $Gamma(\underline{a}_s,\underline{b}_s)$ prior for s_j in constructing the Metropolis-Hastings transition densities $q_s(s_j^{t+1}|m^t,Z^t)$ and $q_m(m_j^{t+1}|s^{t+1},Z^t)$ given correspondingly by

$$Gamma\left(a_j^t+\underline{a}_s-1\ ,\ \left(\frac{1}{b_j^t}+\frac{1}{\underline{b}_s}\right)^{-1}\right) \tag{5}$$

$$Beta\left(n_{t,j}^1+\underline{n}_{m_1}-1,n_{t,j}^0+\underline{n}_{m_0}-1\right), \tag{6}$$

where t is the MCMC iteration index and expressions for $(a_j^t,b_j^t,n_{t,j}^1,n_{t,j}^0)$ are derived from the sampling distribution of the method of moments estimator in Appendix A. We now give a complete description of the MCMC algorithm.

Step 0 : Draw the initial (Z^0,λ^0,s^0,m^0) from the joint prior. Alternatively, one could draw the initial (λ^0,s^0,m^0) from the joint prior and simulate each Z_i^0 independently from the multinomial distribution with parameters $N=1$ and λ^0.

Step 1 : Let (Z^t,λ^t,s^t,m^t) denote draws from the t-th iteration ($t\geq 0$). For all $j\leq M$, draw a candidate for s_j^{t+1} from the proposal density in (5) and denote it by s_j^*. For each j, with probability $\phi_s(s_j^*,s_j^t)$, set $s_j^{t+1}=s_j^*$ and with probability $1-\phi_s(s_j^*,s_j^t)$, reject s_j^* and set $s_j^{t+1}=s_j^t$. The expression for the Metropolis-Hastings acceptance probability $\phi_s(s_j^*,s_j^t)$ is derived in Appendix B.

Step 2 : For each j, draw a candidate for m_j^{t+1} from the proposal density in (6) and denote it by m_j^*. For each j, with probability $\phi_m(m_j^*,m_j^t)$, set $m_j^{t+1}=m_j^*$ and with probability $1-\phi_m(m_j^*,m_j^t)$, reject m_j^* and set $m_j^{t+1}=m_j^t$. The expression for the Metropolis-Hastings acceptance probability $\phi_m(m_j^*,m_j^t)$ is derived in Appendix B.

Step 3 : Note for all k,

$$\pi(Z_i=j|m,s,\lambda,Z_{-i},p)\propto \frac{\lambda_j\Gamma(s_j)p_i^{s_jm_j}(1-p_i)^{s_j(1-m_j)}}{\Gamma(s_jm_j)\Gamma(s_j(1-m_j))} \tag{7}$$

Hence, draw Z_k^{t+1} from a multinomial distribution whose kernel is given by (7) evaluated at $(s_j^{t+1},m_j^{t+1},\lambda_j^t)$.

Step 4 : Note,

$$\pi(\lambda|m,s,Z,p)\propto\prod_j\lambda_j^{\sum_i 1\{Z_i=j\}+\underline{a}-1} \tag{8}$$

Hence, draw λ^{t+1} from $Dirichlet(\sum_i 1\{Z_i^{t+1}=1\}+\underline{a},...,\sum_i 1\{Z_i^{t+1}=M\}+\underline{a})$.

Repeat *Steps 1-5* until convergence is attained.

4 Implementation and performance

Bouguila et al. (2006) use a Metropolis-Hastings random walk (RW) algorithm for transformations of s_j and m_j. We implement both the random walk and the MOM-based independence chain algorithms. The algorithms are programmed in Matlab and the code is available online[3]. The correctness of the algorithms implementation is not rejected by Geweke (2004) joint distribution tests. Both algorithms seem to perform reasonably well for estimation of the function of parameters that are invariant to permutations of the mixture component labels. For a discussion of MCMC and label switching in mixture models see Geweke (2007). The approximations provided by the MOM-based proposals very frequently look almost identical to the target conditional posteriors.

[3] www.princeton.edu/~anorets

To explore the performance of the algorithms further we conduct a Monte Carlo study. First, we generate a 100 draws of (m,s,λ) from a prior. For each draw of the parameters, we generate a dataset of 300 observations and run MOM and RW based algorithms for 100000 iterations. Computing time for one iteration is about the same for both algorithm as most of the time is spent on drawing the latent variables. In the first 10000 iterations, the RW variance parameters are automatically adjusted so that the acceptance rate is close to 50%. The number of mixture components is set to $M = 3$. The prior hyperparameters used in the study are $\underline{n}_{m_1} = \underline{n}_{m_0} = 2$, $\underline{a}_s = 3$, $\underline{b}_s = 100$, and $\underline{a} = 3$.

The algorithms' performance is evaluated by the relative numerical efficiency (RNE).[4] We compute the RNEs for the following permutation invariant objects: $\max m_j$, $\max s_j$, and $\pi(p|m,s,\lambda)$, where p is set to one of the components of the data-generating value of the beta location parameter. The numerical standard errors (the limiting standard deviations of the estimates based on the MCMC draws) necessary for computing the RNEs are obtained by the method of batch means, see Section 4.2 in Tierney (1994).[5]

Table 1 describes the distribution of the ratio of the MOM RNE to the RW RNE for the three objects of interest. Table entries give the frequencies with which the RNE ratios belong to the intervals in the head row of the table. The RNEs were computed from the batches of size 100 (see footnote 5); the results are similar for batch size 1000. The MOM-based algorithm performs better in 90% of the cases. This is not surprising given that the acceptance rates for the MOM independence chain algorithm in most of the simulation experiments were above 0.8 for s_j and above 0.9 for m_j. Obtained efficiency improvement might not matter in simple examples. However, for more complicated hierarchical models, in which a finite beta mixture is used as a flexible prior, such improvements can make an important difference. An example of such a model can be found in Norets and Tang (2010).

Table 1 Distribution of the RNE ratio

% of $\frac{RNE_{MOM}}{RNE_{RW}} \in$	(0,1)	[1,2)	[2,5)	[5,∞)
$\max m_j$	0.11	0.14	0.65	0.1
$\max s_j$	0.1	0.34	0.51	0.05
$\pi(p\|m,s,\lambda)$	0.1	0.38	0.48	0.04

Another advantage of the MOM-based algorithm is that it does not require much tuning.[6] In cases when the extent of the uncertainty about different mixture components is very different, which is likely to happen when the corresponding mixing probabilities are different, tuning the random walk variance parameters might be complicated due to the label switching. Identification restrictions on mixture components such as $m_1 > m_2 > \cdots > m_M$ might make it easier to tune the random walk variances. However, such restrictions are known to slow down mixing in MCMC considerably. Thus, the MOM-based method seems to be an efficient and convenient alternative to the random walk algorithm.

More generally, the proposed approach to construction of the Metropolis-Hastings transition densities can be useful for models in which the MOM estimator is more analytically tractable than the MLE; for example, for models involving gamma distributions, Dirichlet distributions, and their mixtures.

[4] The RNE is defined as the ratio of the variance of a moment estimate based on hypothetical i.i.d. draws to the limiting variance of the estimate based on the MCMC sample. It indicates the number of MCMC draws required to produce the same numerical accuracy as i.i.d. draws directly from the posterior.

[5] Suppose we have MCMC draws $(\theta_1,\ldots,\theta_{L\cdot T})$ and would like to compute the numerical standard error of $\overline{\theta} = \sum_i \theta_i/(LT)$ as an estimator of $E(\theta)$. Divide MCMC draws into T consecutive batches of size L. For each batch j compute the batch mean $\overline{\theta}_j = \sum_{i=(j-1)L}^{jL} \theta_i/L$, $j = 1,\ldots,T$. When the batch size L is large enough, the sequence of batch means can be approximated by an AR(1) process. Thus, the standard error can be approximated by

$$S.E.(\overline{\theta}) \approx \sqrt{\sum(\overline{\theta}_j - \overline{\theta})^2(1+r)/[(1-r)T^2]}$$

where r is the sample auto correlation coefficient for $(\overline{\theta}_1,\ldots,\overline{\theta}_T)$.

[6] In rare cases, the MOM-based algorithm can get stuck if initialized with arbitrary values of parameters and latent variables. In these cases, we use a larger variance for the proposal distribution on initial iterations of the algorithm.

Appenix A: Method of moments estimator and Metropolis-Hastings transition densities

The Metropolis-Hastings transition densities for s_j^{t+1} and m_j^{t+1} are constructed from the sampling distribution of the method of moments estimator of s_j and m_j. The idea is to pick parameters for Gamma and Beta transition densities so that means and variances are equal to the estimated means and variances of the method of moments estimator. For a given dataset p and a previous draw of latent variables Z^t, define $N_j^t = \sum_i 1\{Z_i^t = j\}$.

The method of moment estimator for m_j and its approximate sampling variance are given by

$$\hat{m}_j^t = \sum_{i:Z_i^t=j} p_i/N_j^t \quad \text{and}$$

$$\hat{V}(\hat{m}_j^t) = \sum_{i:Z_i^t=j} (p_i - \hat{m}_j^t)^2/(N_j^t)^2.$$

Conditional on m_j^t (for the Gibbs sampler we need to approximate conditional posterior of s_j given m_j^t), the method of moment estimator for s_j and its approximate sampling variance are given by

$$\hat{s}_j^t = \frac{m_j^t(1-m_j^t)}{(\hat{\sigma}_j^t)^2} - 1 \quad \text{and}$$

$$\hat{V}(\hat{s}_j^t) = \frac{\kappa^4 - (\hat{\sigma}_j^t)^4}{N_j^t(\hat{\sigma}_j^t)^8}(m_j^t)^2(1-m_j^t)^2,$$

where $\kappa^4 = \sum_{i:Z_i^t=j}(p_i - m_j^t)^4/N_j^t$ and

$$(\hat{\sigma}_j^t)^2 = \sum_{i:Z_i^t=j} (p_i - m_j^t)^2/N_j^t.$$

Our choice of proposal densities for s_j^{t+1} and m_j^{t+1} are $Gamma(a_j^t, b_j^t)$ and $Beta(n_{t,j}^1, n_{t,j}^0)$ respectively, with $(a_j^t, b_j^t, n_{t,j}^1, n_{t,j}^0)$ chosen to imply means and variances identical to those estimates of $\hat{s}_j^t$, $\hat{m}_j^t$, $\hat{V}(\hat{m}_j^t)$, and $\hat{V}(\hat{s}_j^t)$ calculated above. Specifically, this amounts to choosing:

$$n_{t,j}^1 = \left[\frac{\hat{m}_j^t(1-\hat{m}_j^t)}{\hat{V}(\hat{m}_j^t)} - 1\right]\hat{m}_j^t,$$

$$n_{t,j}^0 = \left[\frac{\hat{m}_j^t(1-\hat{m}_j^t)}{\hat{V}(\hat{m}_j^t)} - 1\right](1-\hat{m}_j^t),$$

$$a_j^t = \frac{\left(\hat{s}_j^t\right)^2}{\hat{V}(\hat{s}_j^t)},\ b_j^t = \frac{\hat{V}(\hat{s}_j^t)}{\hat{s}_j^t}.$$

Appendix B: Expressions for ϕ_{m_j}, ϕ_{s_j}

The Metropolis-Hastings acceptance probability for drawing s_j^{t+1}, $\phi_{s_j}(s_j^*, s_j^t)$, is given by the minimum of 1 and

$$\frac{\pi(s_j^*, s_{-j}^t, m^t, \lambda^t, Z^t|p)/q_s(s_j^*|m^t, Z^t)}{\pi(s^t, m^t, \lambda^t, Z^t|p)/q_s(s_j^t|m^t, Z^t)},$$

where q_s denotes the proposal density defined in (5). The logarithm of this ratio can be written as

$$\begin{aligned}
&N_j^t\big[\log\Gamma(s_j^*) - \log\Gamma(s_j^t)\\
&+\log\Gamma(s_j^t m_j^t) - \log\Gamma(s_j^* m_j^t)\\
&+\log\Gamma(s_j^t(1-m_j^t)) - \log\Gamma(s_j^*(1-m_j^t))\big]\\
&+m_j^t(s_j^* - s_j^t)\sum_{\{i:Z_i^t=j\}}\log p_i\\
&+(1-m_j^t)(s_j^* - s_j^t)\sum_{\{i:Z_i^t=j\}}\log(1-p_i)\\
&-[(a_j^t-1)(\log s_j^* - \log s_j^t) - (s_j^* - s_j^t)/b_j^t]
\end{aligned}$$

The Metropolis-Hastings acceptance probability for drawing m_j^{t+1}, $\phi_{m_j}(m_j^*, m_j^t)$, is given by the minimum of 1 and

$$\frac{\pi(m_j^*, m_{-j}^t, s^{t+1}, \lambda^t, Z^t|p)/q_m(m_j^*|s^{t+1}, Z^t))}{\pi(m^t, s^{t+1}, \lambda^t, Z^t|p)/q_m(m_j^t|s^{t+1}, Z^t)}$$

where q_m denotes the proposal density defined in (6). The logarithm of this ratio can be written as

$$\begin{aligned}
&N_j^t\big[\log\Gamma(s_j^{t+1}m_j^t) - \log\Gamma(s_j^{t+1}m_j^*)\\
&+\log\Gamma(s_j^{t+1}(1-m_j^t)) - \log\Gamma(s_j^{t+1}(1-m_j^*))\big]\\
&+s_j^{t+1}(m_j^* - m_j^t)\sum_{\{i:Z_i^t=1\}}\log p_i\\
&+s_j^{t+1}(m_j^t - m_j^*)\sum_{\{i:Z_i^t=1\}}\log(1-p_i)\\
&-[(n_{t,j}^1-1)\left(\log m_j^* - \log m_j^t\right)\\
&+(n_{t,j}^0-1)\left(\log(1-m_j^*) - \log(1-m_j^t)\right)].
\end{aligned}$$

References

BOUGUILA, N., D. ZIOU, AND E. MONGA (2006): "Practical Bayesian estimation of a finite beta mixture through gibbs sampling and its applications," *Statistics and Computing*, 16, 215–225.

DIEBOLT, J. AND C. P. ROBERT (1994): "Estimation of Finite Mixture Distributions through Bayesian Sampling," *Journal of the Royal Statistical Society. Series B (Methodological)*, 56, 363–375.

GEWEKE, J. (2004): "Getting it Right: Joint Distribution Tests of Posterior Simulators," *Journal of the American Statsitical Association*, 99, 799–804.

——— (2005): *Contemporary Bayesian Econometrics and Statistics*, Wiley-Interscience.

——— (2007): "Interpretation and inference in mixture models: Simple MCMC works," *Computational Statistics and Data Analysis*, 51, 3529 – 3550.

MCLACHLAN, G. AND D. PEEL (2000): *Finite Mixture Models*, John Wiley & Sons, Inc.

NORETS, A. AND X. TANG (2010): "Semiparametric Inference in Dynamic Binary Choice Models," Unpublished manuscript, Princeton and Upenn.

ROUSSEAU, J. (2010): "Rates of convergence for the posterior distributions of mixtures of betas and adaptive nonparametric estimation of the density," *Ann. Statist.*, 38, 146–180.

TIERNEY, L. (1994): "Markov chains for exploring posterior distributions," *The Annals of Statistics*, 22, 1758–1762.