



Efficient MCMC estimation of discrete distributions

Faming Liang*, Chuanhai Liu

Department of Statistics, Texas A & M University, College Station, Tx 77843 3143, USA,

Received 20 April 2004; received in revised form 27 July 2004; accepted 27 July 2004

Available online 21 August 2004

Abstract

In this paper we propose an efficient Markov chain Monte Carlo (MCMC) method for estimation of discrete distributions by solving an appropriate system of linear equations. We call the estimator the equation-solving estimator. Our numerical results show that the new estimator makes significant improvements over the conventional frequency MCMC estimator in terms of accuracy of the estimates. The new estimator can be used in Bayesian model comparison problems.

© 2004 Elsevier B.V. All rights reserved.

Keywords: Equation solving estimator; Frequency estimator; Markov chain Monte Carlo; Transition matrix

1. Introduction

Let $\pi(\mathbf{x})$, $\mathbf{x} \in \mathcal{X}$ denote the density/mass function of the target distribution, on which we will make inference. Suppose that the sample space \mathcal{X} are partitioned into m subspaces, A_1, \dots, A_m , $\mathcal{X} = \bigcup_{i=1}^m A_i$, $A_i \cap A_j = \emptyset$ for all $i \neq j$, and we are interested in estimating the probabilities

$$\pi(A_i) = \int_{A_i} \pi(\mathbf{x}) \, d\mathbf{x}, \quad i = 1, \dots, m. \quad (1)$$

In Bayesian statistics, many problems can be formulated to calculate the probabilities of the form (1). Here are some examples.

* Corresponding author. Tel.: +979-845-8885; fax: +979-845-3144.

E-mail addresses: fliang@stat.tamu.edu (F. Liang), cliu@stat.tamu.edu (C. Liu).

- (a) *Bayesian hypothesis testing.* Let $m=2$, and A_1 and A_2 denote the subspaces corresponding to the null and alternative hypotheses, respectively. The hypothesis $H_0 : \mathbf{x} \in A_1$ can be tested by comparing the posterior probabilities $\pi(A_1)$ and $\pi(A_2)$.
- (b) *Bayesian model selection.* Let A_i denote the sample space of the model \mathcal{M}_i . Thus, $\pi(A_i)$ is the posterior probability of \mathcal{M}_i , and the best model can be selected by comparing the values of $\pi(A_i)$'s.
- (c) *Change-point identification.* Let A_i denote the space of change-point patterns with i change-points. Thus, $\operatorname{argmax}_{1 \leq i \leq m} \pi(A_i)$ is the most likely number of change-points.

Suppose that a positive recurrent and aperiodic Markov chain with the stationary distribution $\pi(\mathbf{x})$ has been constructed and run, and that a sequence of samples, $\mathbf{x}_1, \dots, \mathbf{x}_n$, has been collected in the simulation. Traditionally, the probability $\pi(A_i)$ is estimated by

$$\widehat{\pi}_n^f(A_i) = \frac{1}{n} \sum_{k=1}^n I(\mathbf{x}_k \in A_i), \quad (2)$$

where $I(\cdot)$ is the indicator function. In this article, we call $\widehat{\pi}_n^f(A_i)$ the frequency estimator of $\pi(A_i)$, since it is the relative frequency of the samples drawn from the subspace A_i . The ergodicity of the Markov chain (Tierney, 1994) implies that

$$\widehat{\pi}_n^f(A_i) \rightarrow \pi(A_i) \quad \text{almost surely,} \quad (3)$$

as the sample size $n \rightarrow \infty$.

We note that the frequency estimator is very general in theory. It can be extended to estimate any expectation $E_\pi h(\mathbf{x})$ with $E_\pi |h(\mathbf{x})| < \infty$, while keeping the ergodicity property (3) held. However, it may not be optimal for estimating quantities of form (1) in terms of accuracy of the estimates. In this paper, we propose a new method for estimating the distribution $\pi(A_i)$ by constructing an estimate of a probability transition matrix with $\pi(A_i)$ as its stationary distribution and solving the corresponding system of linear equations with $\pi(A_i)$ as its solution. The new method leads to more accurate estimates than the frequency method in all numerical examples of this article.

The remaining part of this paper is organized as follows. In Section 2, we describe the new estimator and explore its asymptotic property. In Section 3, we give an illustrative example. In Section 4, we consider the application of the new estimator in estimating ratios of normalizing constants of a sequence of distributions. In Section 5, we consider the application of the new estimator in the problem of change-point identification. In Section 6, we conclude the paper with a brief discussion.

2. The equation solving estimator

For a Markov chain with the state space $\{1, \dots, m\}$, the standard Markov chain theory (Roberts, 1996) states that if the Markov chain is positive recurrent and aperiodic, then its stationary distribution $\pi(\cdot)$ is the unique probability distribution satisfying the

following equation:

$$\sum_{i=1}^m \pi(i) P_{ij}(k) = \pi(j) \tag{4}$$

for all $j = 1, \dots, m$ and $k > 0$, where $P_{ij}(k) = P[X_k = j | X_0 = i]$ is the k -step transition probability for $X_0 = i$ and $X_k = j$. Therefore, if the transition matrix $(P_{ij}(k))_{m \times m}$ is known for a given value of k , say, $k = 1$, $\pi(i)$ can be obtained by solving Eq. (4). In this case, the estimator of $\pi(i)$ has zero variance. If this is not the case, the transition matrix can still be accurately estimated because we can take advantage of some underlying probability calculations in MCMC simulation. Hence, we can use the probabilities solving from (4) instead of the empirical 0/1 MC occurrences for estimation.

Bearing this idea in mind, we propose the following scheme to estimate the transition matrix and then to solve the equation for $\pi(i)$'s. For simplicity, henceforth, we denote by $P = (P_{ij}(1))_{m \times m}$ the one-step transition matrix of the Markov chain. Suppose that the Metropolis–Hastings algorithm (Metropolis et al., 1953; Hastings, 1970) is used to simulate a sequence of $\{\mathbf{x}_t\}$ from the target distribution $\pi(\mathbf{x})$. Let \tilde{P} denote a matrix which cumulates the one-step transition probabilities calculated in the simulation, and \mathbf{x}_0 denote the initial state of the Markov chain. Set $t = 0$ and $\tilde{P} = \mathbf{0}$. The algorithm iterates between the following steps.

- (a) Draw a random sample from the proposal distribution $T(\mathbf{y}|\mathbf{x}_t)$.
- (b) Calculate the transition ratio

$$\alpha(\mathbf{x}_t, \mathbf{y}) = \frac{\pi(\mathbf{y})}{\pi(\mathbf{x}_t)} \frac{T(\mathbf{x}_t|\mathbf{y})}{T(\mathbf{y}|\mathbf{x}_t)}. \tag{5}$$

Set $\mathbf{x}_{t+1} = \mathbf{y}$ with the probability $\min\{1, \alpha(\mathbf{x}_t, \mathbf{y})\}$ and set $\mathbf{x}_{t+1} = \mathbf{x}_t$ with the remaining probability.

- (c) Suppose $\mathbf{x}_t \in A_i$ and $\mathbf{y} \in A_j$, set $\tilde{P}_{ij} \leftarrow \tilde{P}_{ij} + \frac{\alpha(\mathbf{x}_t, \mathbf{y})}{1 + \alpha(\mathbf{x}_t, \mathbf{y})}$ and $\tilde{P}_{ii} \leftarrow \tilde{P}_{ii} + \frac{1}{1 + \alpha(\mathbf{x}_t, \mathbf{y})}$.

At the end of iterations, we normalize \tilde{P} so that each row of the normalized matrix sums to 1 and denote the normalized matrix by $\hat{P}^{(B)}$. We note that the estimator $\hat{P}^{(B)}$ corresponds to the Boltzmann algorithm (Winkler, 1995) of the Markov chain construction. The element $\hat{P}_{ij}^{(B)}$ estimates the transition probability for a sample from A_i to A_j in one transition step, that is,

$$P(A_i \rightarrow A_j) = \frac{1}{\int_{A_i} \pi(\mathbf{x}) d\mathbf{x}} \int_{A_i} \pi(\mathbf{x}) \left[\int_{A_j} K(\mathbf{y}|\mathbf{x}) d\mathbf{y} + r(\mathbf{x}) I(\mathbf{x} \in A_j) \right] d\mathbf{x},$$

where $K(\mathbf{y}|\mathbf{x}) = T(\mathbf{y}|\mathbf{x})\alpha(\mathbf{x}, \mathbf{y})$ is the transition kernel density, and $r(\mathbf{x}) = 1 - \int K(\mathbf{y}|\mathbf{x}) d\mathbf{y}$ is the rejection probability of a transition from point \mathbf{x} .

Given $\widehat{P}^{(B)}$, $\pi(A_i)$'s can then be estimated by solving the system of linear equations,

$$\sum_{i=1}^m \widehat{\pi}_n^e(A_i) \widehat{P}_{ij}^{(B)} = \widehat{\pi}_n^e(A_j), \quad j = 1, \dots, m$$

subject to
$$\sum_{i=1}^m \widehat{\pi}_n^e(A_i) = 1, \quad (6)$$

where the subscript n denotes the number of iterations performed in the MCMC simulation. We call $\widehat{\pi}_n^e(A_i)$ the equation solving estimator of $\pi(A_i)$. Solving the constrained Eq. (6) is equivalent to solving the following unconstrained linear system

$$\begin{aligned} (Q - bs' - I)\widehat{\pi}_{n,-m}^e &= -b, \\ \widehat{\pi}_n^e(A_m) &= 1 - s'\widehat{\pi}_{n,-m}^e, \end{aligned} \quad (7)$$

where $b = (\widehat{P}_{1,m}^{(B)}, \dots, \widehat{P}_{m-1,m}^{(B)})'$, s is a $(m-1)$ -vector of 1's, $\widehat{\pi}_{n,-m}^e = (\widehat{\pi}_n^e(A_1), \dots, \widehat{\pi}_n^e(A_{m-1}))$, and Q is the left-upper $(m-1) \times (m-1)$ submatrix of $\widehat{P}^{(B)}$, that is,

$$Q = \begin{pmatrix} \widehat{P}_{1,1}^{(B)} & \widehat{P}_{1,2}^{(B)} & \cdots & \widehat{P}_{1,m-1}^{(B)} \\ \widehat{P}_{2,1}^{(B)} & \widehat{P}_{2,2}^{(B)} & \cdots & \widehat{P}_{2,m-1}^{(B)} \\ \vdots & \vdots & \vdots & \vdots \\ \widehat{P}_{m-1,1}^{(B)} & \widehat{P}_{m-1,2}^{(B)} & \cdots & \widehat{P}_{m-1,m-1}^{(B)} \end{pmatrix}.$$

Due to the non-uniqueness of the acceptance rule of the Markov chain (Hastings, 1970), the transition matrix can be estimated alternatively as follows.

- (c)' If $\alpha(\mathbf{x}_t, \mathbf{y}) > 1$, set $\tilde{P}_{ij} \leftarrow \tilde{P}_{ij} + 1$. Otherwise, set $\tilde{P}_{ij} \leftarrow \tilde{P}_{ij} + \alpha(\mathbf{x}_t, \mathbf{y})$ and $\tilde{P}_{ii} \leftarrow \tilde{P}_{ii} + 1 - \alpha(\mathbf{x}_t, \mathbf{y})$.

We denote the resulting transition matrix estimator by $\widehat{P}^{(M)}$. Henceforth, we will denote by $\hat{\pi}_{n,B}^e$ and $\hat{\pi}_{n,M}^e$ the derived estimators resulted from $\widehat{P}^{(B)}$ and $\widehat{P}^{(M)}$, respectively. In Section 3, we show that $\hat{\pi}_{n,B}^e$ is more efficient than $\hat{\pi}_{n,M}^e$ by a numerical example.

The following theorem states the asymptotic property of the equation solving estimator $\widehat{\pi}_n^e(\cdot)$. The proof is presented in Appendix.

Theorem 2.1. *The equation solving estimator $\widehat{\pi}_n^e(\cdot)$ converges to $\pi(\cdot)$ almost surely as the number of iterations used for estimating the transition matrix tends to infinity, that is,*

$$\widehat{\pi}_n^e(A_i) \rightarrow \pi(A_i), \quad \text{almost surely}, \quad (8)$$

as $n \rightarrow \infty$.

Comparing (3) and (8), we know that both the frequency estimator and the equation solving estimator are consistent for $\pi(\cdot)$. However, in practice, the equation solving estimator usually converges faster than the frequency estimator. This is because the randomness of selection between \mathbf{x}_t and \mathbf{y} has been integrated out when we estimate the

transition matrix in MCMC simulations. A theoretical justification is the theory about the Rao–Blackwellisation theorem and the theory of the commonly used δ -method in investigation of asymptotic results. The Rao–Blackwellisation theorem encourages one to do as much analytical work as possible in Monte Carlo computation based on the following inequality:

$$\text{Var}(E(W(\mathbf{x}|\tilde{\mathbf{x}}^{(2)}))) \leq \text{Var}(W(\mathbf{x})), \tag{9}$$

where $\tilde{\mathbf{x}}^{(2)}$ is a subvector of \mathbf{x} , that is, $\mathbf{x} = (\tilde{\mathbf{x}}^{(1)}, \tilde{\mathbf{x}}^{(2)})$; $W(\mathbf{x})$ denotes an estimator of a quantity; and $E(W(\mathbf{x})|\tilde{\mathbf{x}}^{(2)})$ is the conditional expectation of $W(\mathbf{x})$ conditioning on $\tilde{\mathbf{x}}^{(2)}$. Note (9) implies that the estimator $\frac{1}{N} \sum_{i=1}^N E W(\mathbf{x}|\tilde{\mathbf{x}}_i^{(2)})$ has a smaller variance than $\frac{1}{N} \sum_{i=1}^N W(\mathbf{x}_i)$, where $\mathbf{x}_1, \dots, \mathbf{x}_N$ are iid samples drawn from the target distribution. For more discussions on the theorem see Casella and Roberts (1996).

3. An illustrative example

We illustrate the equation solving estimator by a Bayesian hypothesis testing problem, which is to test if a coefficient of a logistic regression is less than zero or not.

The dataset we considered is taken from Ashford (1959). It concerns the proportion of coal miners who exhibit symptoms of severe pneumoconiosis and the number of years of exposure. We follow (Montgomery et al., 2001) to fit a logistic regression model

$$E(y_i) = \frac{\exp(\beta_0 + \beta_1 z_i + \beta_2 z_i^2)}{1 + \exp(\beta_0 + \beta_1 z_i + \beta_2 z_i^2)}, \quad i = 1, \dots, N,$$

to the data, where z_i is the number of years of exposure of the i th miner, y_i is the binary indicator of severe pneumoconiosis, and β_i 's are the regression coefficients, and N is the number of observations. By assuming that β_0, β_1 and β_2 are independent a priori and follow a common normal distribution with the mean 0 and variance $\tau^2 = 100$, we have the following log-posterior density (up to an additive constant),

$$\begin{aligned} \log \pi(\boldsymbol{\beta}|D) = & \sum_{i=1}^N y_i(\beta_0 + \beta_1 z_i + \beta_2 z_i^2) - \sum_{i=1}^n \log(1 + \exp(\beta_0 + \beta_1 z_i + \beta_2 z_i^2)) \\ & - \frac{1}{2\tau^2} (\beta_0^2 + \beta_1^2 + \beta_2^2), \end{aligned}$$

where $\boldsymbol{\beta} = (\beta_0, \beta_1, \beta_2)$. For this example, testing the hypothesis $H_0 : \beta_2 < 0$ is reduced to comparing the posterior probabilities $\pi(A_1)$ and $\pi(A_2)$, where $A_1 = \{\boldsymbol{\beta} : \beta_2 < 0\}$ and $A_2 = \{\boldsymbol{\beta} : \beta_2 \geq 0\}$.

To simulate from the posterior, we applied the Metropolis–Hastings algorithm with the proposal distribution $\boldsymbol{\beta}_{t+1} \sim N_3(\boldsymbol{\beta}_t, \text{diag}[\sigma_1^2, \sigma_2^2, \sigma_3^2])$, where $\boldsymbol{\beta}_t$ is the estimate of $\boldsymbol{\beta}$ obtained at the t th iteration, and $\sigma_1 = 1, \sigma_2 = 0.1$ and $\sigma_3 = 0.01$. Here we choose different step-sizes for β_0, β_1 and β_2 to accommodate different scales of the corresponding “explanatory variables”. The simulation consists of 100 replications. Each run consists of 110000 iterations, where the first 10000 iterations was discarded for the burn-in process, and the

Table 1
Comparison of the three estimators $\hat{\pi}_n^f$, $\hat{\pi}_{n,B}^e$ and $\hat{\pi}_{n,M}^e$

Prob.	Frequency		Equation solving (scheme B)			Equation solving (scheme M)		
	Estimate (%)	SD ($\times 10^{-3}$)	Estimate (%)	SD ($\times 10^{-4}$)	Saving (%)	Estimate (%)	SD ($\times 10^{-4}$)	Saving (%)
$\pi(A_1)$	95.62	2.523	95.77	2.120	41.6	95.76	2.178	34.2
$\pi(A_2)$	4.38	2.523	4.23	2.120	41.6	4.24	2.178	34.2

The “Estimate” was calculated by averaging over 100 runs, SD was the standard deviation of “Estimate”, and the “Saving” is defined as $1 - [\text{SD}(\hat{\pi}_{n,S}^e)/\text{SD}(\hat{\pi}_n^f)]^2$, where $S \in \{B, M\}$ denotes the estimation scheme.

remaining 100000 iterations was used for estimation. The following are some numerical results produced by a typical run. The estimate of β is $(-6.7508, 0.2256, -0.0020)$, which is rather close to its maximum likelihood estimate (MLE) $(-6.7108, 0.2276, -0.0021)$ given in Montgomery et al. (2001). This indicates the our implementation is effective for simulating from the posterior $\pi(\beta|D)$. The unnormalized transition matrix produced in the run is

$$\tilde{P}^{(B)} = \begin{pmatrix} 95156.1 & 10.9152 \\ 12.682 & 4820.32 \end{pmatrix}.$$

Normalizing $\tilde{P}^{(B)}$ and solving Eq. (6), we got the estimate $\hat{\pi}_{n,B}^e = (0.9581, 0.0419)$. The visiting frequencies to the subspaces A_1 and A_2 were also recorded in the run, which are 95167 and 4833, respectively. The resulting frequency estimate is $(0.9517, 0.0483)$. Note that other than the visiting frequencies to each of the subspaces (the sum of the i th row of $\tilde{P}^{(B)}$ is equal to the visiting frequency to the subspace A_i), $\tilde{P}^{(B)}$ records more information on the simulation, the number of transitions between the subspaces. We expect that the use of the extra information will lead to an estimator which outperforms the frequency estimator.

In addition to $\hat{\pi}_{n,B}^e$ and $\hat{\pi}_n^f$, $\hat{\pi}_{n,M}^e$ was also calculated in each run. Table 1 summarizes the results of the 100 runs. The results show that both of the two equation solving estimators have made significant improvements over the frequency estimator in terms of accuracy. The savings are more than 30% for this example. Although the improvements are not drastic, the extra cost for the improvement is almost negligible in terms of CPU time, just solving a linear system. Table 1 shows that the two equation solving estimators have almost the same performance. Taking a closer look at the table, we may find that $\hat{\pi}_{n,B}^e$ is slightly better than $\hat{\pi}_{n,M}^e$. This is because each update of $\tilde{P}^{(B)}$ is more moderate than that of $\tilde{P}^{(M)}$. The values to add for $\tilde{P}^{(B)}$ are always between 0 and 1, while they may be 1 for $\tilde{P}^{(M)}$. The extremeness of the modification deteriorates the performance of $\hat{\pi}_{n,M}^e$. In the following examples, $\hat{\pi}_{n,M}^e$ will be omitted, and only $\hat{\pi}_{n,B}^e$ will be reported.

4. Estimating ratios of normalizing constants of different distributions

Suppose we are interested in estimating the ratios of the normalizing constants of m distributions,

$$\pi_i(\mathbf{x}) = \frac{1}{Z_i} g(\mathbf{x})^{\beta_i}, \quad i = 1, \dots, m,$$

where β_i is the inverse of the temperature. Here we set $m = 5$, $(1/\beta_1, \dots, 1/\beta_5) = (8, 4, 2, 1, 0.5)$, $\mathbf{x} = (x_1, x_2)$ is a two-dimensional vector, and

$$g(\mathbf{x}) = \frac{1}{4\pi} \{e^{-\frac{1}{2} \sum_{j=1}^2 (x_j-5)^2} + e^{-\frac{1}{2} \sum_{j=1}^2 (x_j+5)^2}\}.$$

This example mimics a Bayesian model selection problem.

To estimate the ratios of these normalizing constants, we construct the following MCMC scheme, which works in the style of simulated tempering (Marinari and Parisi, 1992; Geyer and Thompson, 1995) by simulating from the joint distribution

$$\pi(\mathbf{x}, t) = \sum_{i=1}^m g(\mathbf{x})^{\beta_i} I(t = i),$$

where $I(\cdot)$ is the indicator function and $t \in \{1, \dots, m\}$ indicates the level of the temperature of \mathbf{x} . Let \mathbf{x}_k denote the state of the Markov chain at the k th iteration, and $t(k) \in \{1, \dots, m\}$ denote the level of the temperature associated with \mathbf{x}_k , that is, $\mathbf{x}_k \sim \pi_{t(k)}(\mathbf{x})$. To estimate the transition matrix P , we perform n iterations of the following steps.

- (1) Draw a random number $U \sim \text{Unif}(0, 1)$.
- (2) If $U < q_{t(k), t(k)-1}$, update $t(k)$ by a move-down procedure; otherwise, if $U < q_{t(k), t(k)+1}$, update \mathbf{x}_k by a sample-update procedure; otherwise, update $t(k)$ by a move-up procedure.

In our simulation, we set $q_{i,i-1} = q_{i,i} = q_{i,i+1} = 1/3$ for $2 \leq i \leq m-1$, $q_{1,0} = q_{m,m+1} = 0$, $q_{1,2} = q_{m,m-1} = \frac{1}{3}$, and $q_{1,1} = q_{m,m} = \frac{2}{3}$. In the move-down procedure, we accept the change of the temperature with probability $\min\{1, r_d\}$, where

$$r_d = \frac{\pi_{t(k)-1}(\mathbf{x}_k) q_{t(k)-1, t(k)}}{\pi_{t(k)}(\mathbf{x}_k) q_{t(k), t(k)-1}}. \tag{10}$$

If it is accepted, we set $\mathbf{x}_{k+1} = \mathbf{x}_k$ and $t(k+1) = t(k) - 1$, otherwise, we set $\mathbf{x}_{k+1} = \mathbf{x}_k$ and $t(k+1) = t(k)$. Set $\tilde{P}_{t(k), t(k)-1} \leftarrow \tilde{P}_{t(k), t(k)-1} + \frac{r_d}{1+r_d}$ and $\tilde{P}_{t(k), t(k)} \leftarrow \tilde{P}_{t(k), t(k)} + \frac{1}{1+r_d}$.

In the move-up procedure, we accept the change of the temperature with probability $\min\{1, r_u\}$, where

$$r_u = \frac{\pi_{t(k)+1}(\mathbf{x}_k) q_{t(k)+1, t(k)}}{\pi_{t(k)}(\mathbf{x}_k) q_{t(k), t(k)+1}}. \tag{11}$$

If it is accepted, we set $\mathbf{x}_{k+1} = \mathbf{x}_k$ and $t(k+1) = t(k) + 1$, otherwise, we set $\mathbf{x}_{k+1} = \mathbf{x}_k$ and $t(k+1) = t(k)$. Set $\tilde{P}_{t(k), t(k)+1} \leftarrow \tilde{P}_{t(k), t(k)+1} + \frac{r_u}{1+r_u}$ and $\tilde{P}_{t(k), t(k)} \leftarrow \tilde{P}_{t(k), t(k)} + \frac{1}{1+r_u}$.

Table 2

Comparison of the frequency estimator and the equation solving estimator for estimating ratios of normalizing constants of different distributions

	Frequency		Equation solving		
	Estimate	SD($\times 10^{-2}$)	Estimate	SD($\times 10^{-2}$)	Saving (%)
Z_1	67.753	1.870	67.748	1.587	28.0
Z_2	24.786	1.235	24.773	1.019	32.0
Z_3	6.521	0.881	6.534	0.707	35.7
Z_4	0.903	0.323	0.910	0.255	37.7
Z_5	0.037	0.062	0.035	0.043	52.6

The frequency estimator is defined as $\widehat{Z}_i = \frac{\#\{k:t(k)=i\}}{n} \times 100$ for this example. “Estimate” was calculated by averaging over 500 runs. “SD” is the standard deviation of “Estimate”. “Saving” was computed as in Table 1.

In the sample-update procedure, we first propose a new sample $\mathbf{y} = \mathbf{x}_k + z\mathbf{e}$, where \mathbf{e} is a uniform direction in the two-dimensional space and $z \sim N(0, 1/\beta_{t(k)})$. The new sample is accepted with probability r_m , where

$$r_m = \frac{\pi_{t(k)}(\mathbf{y})}{\pi_{t(k)}(\mathbf{x}_k)}.$$

If it is accepted, we set $\mathbf{x}_{k+1} = \mathbf{y}$ and $t(k+1) = t(k)$, otherwise, we set $\mathbf{x}_{k+1} = \mathbf{x}_k$ and $t(k+1) = t(k)$. Set $\tilde{P}_{t(k),t(k)} \leftarrow \tilde{P}_{t(k),t(k)} + 1$.

The algorithm was run for 500 times independently. Each run consists of 100000 iterations and costs about 0.13s on a 2.8 GHZ computer (all computations of this article were done on the same computer). Table 2 shows the relative efficiency of the frequency estimator over that of the equation solving estimator. It shows again that the equation solving estimator has made a significant improvement over the frequency estimator.

5. Change point identification

The problem of change-point identification can be stated as follows. Given a sequence of observations $D = (z_1, z_2, \dots, z_n)$, which are ordered in time. We assume that the z_i 's are independent, and there exists a partition on the set $\{1, 2, \dots, n\}$ into blocks such that the sequence follows the same distribution within blocks, more specifically, there exists a binary vector $\mathbf{x} = (x_1, x_2, \dots, x_{n-1})$ with $x_{c_1} = \dots = x_{c_k} = 1$ and being 0 elsewhere, where

$$0 = c_0 < c_1 < \dots < c_k < c_{k+1} = n$$

and

$$z_i \sim \pi_r(\cdot), \quad c_{r-1} < i \leq c_r$$

for $r = 1, 2, \dots, k+1$. For simplicity, we further assume that $\pi_r(\cdot)$ is a density determined by a set of parameters $\theta_r \in \Theta$. The parameters change at points $c_1 + 1, \dots, c_k + 1$. The task is to identify the unknown values of c_1, \dots, c_k .

This problem has been studied recently by several authors using simulation-based methods, such as the Gibbs sampler (Barry and Hartigan, 1993), jump diffusion (Philips and Smith, 1996), reversible jump MCMC (Green, 1995) and evolutionary Monte Carlo (Liang and Wong, 2000). In this article, we follow Barry and Hartigan (1993) to consider the case where $\pi_r(\cdot)$ is a Gaussian density parameterized by $\theta_r = (\mu_r, \sigma_r^2)$. Let $\mathbf{x}^{(k)}$ denote a configuration of \mathbf{x} with k ones, which represents a model with k change points. Let $\eta^{(k)} = (\mathbf{x}^{(k)}, \mu_1, \sigma_1^2, \dots, \mu_{k+1}, \sigma_{k+1}^2)$, and denote by A_k the space of models with k change points, $\mathbf{x}^{(k)} \in A_k$, and $\mathcal{X} = \cup_{k=0}^n A_k$. The log-likelihood of model $\eta^{(k)}$ is

$$\log \pi(D|\eta^{(k)}) = - \sum_{i=1}^{k+1} \left\{ \frac{c_i - c_{i-1}}{2} \log \sigma_i^2 + \frac{1}{2\sigma_i^2} \sum_{j=c_{i-1}+1}^{c_i} (z_j - \mu_i)^2 \right\}. \tag{12}$$

For a Bayesian analysis, we use the following prior distributions for $\eta^{(k)}$:

$$\pi(\mathbf{x}^{(k)}) = \frac{\lambda^k}{\sum_{j=0}^{n-1} \frac{\lambda^j}{j!}} \frac{(n-1-k)!}{(n-1)!}, \quad k = 0, 1, \dots, n-1.$$

This is equivalent to assuming that A_k has a truncated Poisson distribution with parameter λ , and each of the $(n-1)!/[k!(n-1-k)!]$ models in A_k is equally likely *a priori*. We put an improper prior on μ_i 's and an inverse-Gamma $IG(\gamma, \delta)$ on σ_i^2 's. Assuming that all the priors are independent, we have the log-prior density (up to an additive constant),

$$\log \pi(\eta^{(k)}) = a_k - \sum_{i=1}^{k+1} \left[(\gamma - 1) \log \sigma_i^2 + \frac{\delta}{\sigma_i^2} \right], \tag{13}$$

where $a_k = (k+1)[\gamma \log \delta - \log \Gamma(\gamma)] + \log(n-1-k)! + k \log \lambda$. The γ, δ and λ are fixed hyperparameters. The log-posterior of $\eta^{(k)}$ (up to an additive constant) can be obtained by adding (12) and (13). Integrating out $\mu_1, \sigma_1^2, \dots, \mu_{k+1}, \sigma_{k+1}^2$ from the full posterior distribution and taking a logarithm, we have

$$\begin{aligned} \log \pi(\mathbf{x}^{(k)}|D) = a_k + \frac{k+1}{2} \log 2\pi - \sum_{i=1}^{k+1} & \left\{ \frac{1}{2} \log(c_i - c_{i-1}) \right. \\ & - \log \Gamma \left(\frac{c_i - c_{i-1} - 1}{2} + \gamma \right) + \left(\frac{c_i - c_{i-1} - 1}{2} + \gamma \right) \\ & \left. \times \log \left[\delta + \frac{1}{2} \sum_{j=c_{i-1}+1}^{c_i} z_j^2 - \frac{\left(\sum_{j=c_{i-1}+1}^{c_i} z_j \right)^2}{2(c_i - c_{i-1})} \right] \right\}. \end{aligned} \tag{14}$$

In addition to identifying the change points, we are often interested in estimating the probabilities $P(A_k|D)$ for $k_{\min} \leq k \leq k_{\max}$, where k_{\min} and k_{\max} specify the range of the number of change-points of interest.

To estimate the transition matrix which comprises the transition probabilities between different A_k 's, we have the following algorithm. The algorithm starts with $\hat{P} = \mathbf{0}$ and a

random configuration $\mathbf{x}^{(k)}$ with $k_{\min} \leq k \leq k_{\max}$, and then iterates between the following steps.

- Set $j = k - 1, k$, or $k + 1$ according to the probabilities $q_{k,j}$, where $q_{k,k} = \frac{1}{3}$ for $k_{\min} \leq k \leq k_{\max}$, $q_{k_{\min}, k_{\min}+1} = q_{k_{\max}, k_{\max}-1} = \frac{2}{3}$, and $q_{k,k+1} = q_{k,k-1} = \frac{1}{3}$ if $k_{\min} < k < k_{\max}$.
- If $j = k$, update $\mathbf{x}^{(k)}$ with a “simultaneous” move (described below); if $j = k + 1$, update $\mathbf{x}^{(k)}$ with a “birth” move (described below); and if $j = k - 1$, update $\mathbf{x}^{(k)}$ with a “death” move (described below).

The “birth”, “death”, and “simultaneous” moves are similar to that described in Green (1995). In the “birth” move, a random number, say u , is first drawn uniformly from the set $\{0, 1, \dots, k\}$; then another random number, say v , is drawn uniformly from the set $\{c_u + 1, \dots, c_{u+1} - 1\}$, and it is proposed to set $x_{i,v} = 1$. The resulting new sample is denoted by $\mathbf{x}_*^{(k+1)}$. In the “death” move, a random number, say u , is drawn uniformly from the set $\{1, 2, \dots, k\}$, and it is proposed to set $x_{i,c_u} = 0$. The resulting new sample is denoted by $\mathbf{x}_*^{(k-1)}$. In the “simultaneous” move, a random number, say u , is first randomly drawn from the set $\{1, 2, \dots, k\}$; then another random number, say v , is uniformly drawn from the set $\{c_{u-1} + 1, \dots, c_u - 1, c_u + 1, \dots, c_{u+1} - 1\}$, and it is proposed to set $x_{i,c_u} = 0$ and $x_{i,v} = 1$. The resulting new sample is denoted by $\mathbf{x}_*^{(k)}$. The acceptance probabilities of the three types of moves are as follows. For the “birth” move, it is $\min(1, r_b)$, where

$$r_b = \frac{\pi(\mathbf{x}_*^{(k+1)} | D)}{\pi(\mathbf{x}^{(k)} | D)} \frac{q_{k+1,k}}{q_{k,k+1}} \frac{c_{u+1} - c_u - 1}{1}. \quad (15)$$

Set $\tilde{P}_{k,k+1} \leftarrow \tilde{P}_{k,k+1} + \frac{r_b}{1+r_b}$ and $\tilde{P}_{k,k} \leftarrow \tilde{P}_{k,k} + \frac{1}{1+r_b}$. For the “death” move, it is $\min(1, r_d)$, where

$$r_d = \frac{\pi(\mathbf{x}_*^{(k-1)} | D)}{\pi(\mathbf{x}^{(k)} | D)} \frac{q_{k-1,k}}{q_{k,k-1}} \frac{1}{c_{u+1} - c_{u-1} - 1}. \quad (16)$$

Set $\tilde{P}_{k,k-1} \leftarrow \tilde{P}_{k,k-1} + \frac{r_d}{1+r_d}$ and $\tilde{P}_{k,k} \leftarrow \tilde{P}_{k,k} + \frac{1}{1+r_d}$. For the “simultaneous” move, it is $\min(1, r_s)$, where

$$r_s = \frac{\pi(\mathbf{x}_*^{(k)} | D)}{\pi(\mathbf{x}^{(k)} | D)}. \quad (17)$$

For this move type, the proposal densities are symmetric in the sense that $T(\mathbf{x}^{(k)} \rightarrow \mathbf{x}_*^{(k)}) = T(\mathbf{x}_*^{(k)} \rightarrow \mathbf{x}^{(k)}) = 1/(c_{u+1} - c_{u-1} - 2)$. Set $\tilde{P}_{k,k} \leftarrow \tilde{P}_{k,k} + 1$.

5.1. A simulation study

The simulated dataset consists of 200 observations with $z_1, \dots, z_{40} \sim N(0, 1)$, $z_{41}, \dots, z_{70} \sim N(1.5, 1)$, $z_{71}, \dots, z_{120} \sim N(-0.5, 2)$, $z_{121}, \dots, z_{180} \sim N(1, 1)$, and $z_{181}, \dots, z_{200} \sim N(0, 2)$. The time series plot of the simulated sequence is shown in Fig. 1. In simulation, we set $\gamma = \delta = 0.05$, $\lambda = 1$, $k_{\min} = 3$, and $k_{\max} = 7$. The algorithm was then run for 50 times

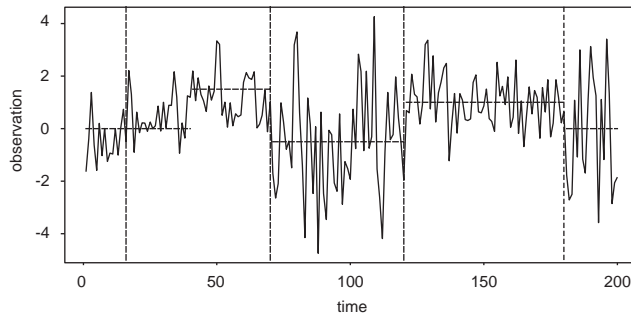


Fig. 1. Comparison of the true (horizontal lines) and MAP estimate (vertical lines) of the change points.

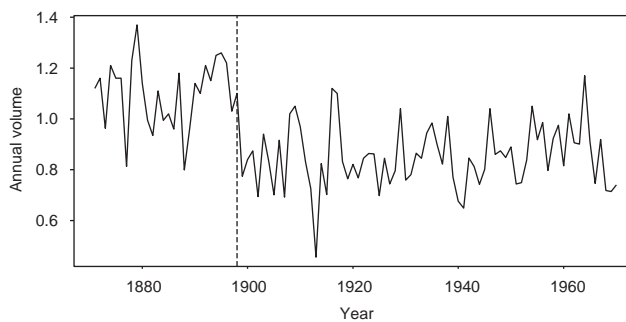


Fig. 2. Time series of the annual volume of the Nile River (discharge at Aswan, 10^{11} m^3) from 1871 to 1970. The dotted line shows the change-point position identified by the simulation.

independently. Each run consists of 10^6 iterations and costs about 27s. The overall acceptance rate of the birth and death moves is about 0.15. Fig. 1 compares the true and MAP estimates of the change points. The MAP estimate of the change-point pattern is (16,70,120,180). It is quite consistent with the true pattern (40,70,120,180). The discrepancy of the two patterns happens at the first change-point position. A detailed exploration of the original data gives a strong support to our estimate. The last 24 observations of the first cluster have a larger mean value than that expected, and they tend to be clustered into the second cluster. We note that the log-posterior probability of the MAP pattern is of 2.03 higher than that of the true pattern.

For this example, we also compare the efficiency of the frequency estimator and the equation solving estimator by comparing standard deviations of the estimates of $\pi(A_k)$'s. The saving created by the equation solving estimator is about 20% in terms of CPU time. The saving is computed as in Table 1.

5.2. Nile discharge

Here we consider a real data problem. The dataset (shown in Fig. 2) is on the annual volume of discharge on the Nile river at Aswan for the years 1871–1970. It has been

analyzed as a change-point identification problem by a number of authors, including Cobb (1978), Freeman (1986), and Phillips and Smith (1996). For this dataset we set $\gamma = \delta = 0.05$ and $\lambda = 1$, which are the same as that used for the simulated example. In simulation, we set $k_{\min} = 1$ and $k_{\max} = 2$. The algorithm was run for 100 times independently. Each run consists of 10000 iterations and costs about 0.15 s. The MAP estimate of the change pattern is shown in Fig. 2. For this example, the saving created by the equation solving estimator is about 40%.

6. Conclusion

We proposed an equation solving MCMC estimator for estimating discrete distributions. Our numerical results show that the new estimator has made a significant improvement over the conventional frequency estimator in terms of accuracy of the estimates. The improvement is consistent over all numerical examples we have studied.

The key step of the equation solving estimator is to estimate the transition matrix \hat{P} , which requires that the transition ratio $\alpha(\mathbf{x}, \mathbf{y})$ (Eq. (5)) must be calculable for any two points \mathbf{x} and \mathbf{y} in the space \mathcal{X} . This is true in general. Even for some MCMC algorithms, say, simulated tempering (Marinari and Parisi, 1992; Geyer and Thompson, 1995), the transition ratio associated directly with the simulation is not calculable, the transition matrix can still be estimated by specifying a virtual proposal distribution. In simulated tempering, the simulation is made for the joint distribution $\pi(\mathbf{x}, \beta)$ of \mathbf{x} and an auxiliary variable β , where β usually refers to the inverse of the temperature, the transition proposal $T(\mathbf{x} \rightarrow \mathbf{y})$ is not calculable due to that the corresponding path integral (Schulman, 1981) is intractable analytically. The path integral evaluates the transition probability of a particle from one point to another in a time-evolution process. In this case, the transition matrix can be estimated by specifying a virtual proposal distribution $T^*(\mathbf{x} \rightarrow \mathbf{y})$ defined on the space \mathcal{X} . The $T^*(\mathbf{x} \rightarrow \mathbf{y})$ is called the virtual proposal distribution as it is not really used for simulation from $\pi(\cdot)$. Let $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ denote a pool of existing samples which are collected in the simulation. The \hat{P} can then be calculated as in Section 2 except that the transition ratio is replaced by

$$\alpha^*(\mathbf{x}_i, \mathbf{y}^*) = \frac{\pi(\mathbf{y}^*)}{\pi(\mathbf{x}_i)} \frac{T^*(\mathbf{y}^* \rightarrow \mathbf{x}_i)}{T^*(\mathbf{x}_i \rightarrow \mathbf{y}^*)}, \quad i = 1, \dots, n, \quad (18)$$

where \mathbf{y}^* denotes a sample resampled from the pool \mathbf{X} . In resampling the \mathbf{y}^* used in (18), each sample \mathbf{x}_j is assigned a weight $w_j = T^*(\mathbf{x}_i \rightarrow \mathbf{x}_j) / \pi(\mathbf{x}_j)$. Hence, the equation solving estimator can be used with general MCMC algorithms or existing samples to improve the accuracy of estimation.

The value of the equation solving estimator can be justified as follows. First, it outperforms the frequency estimator at a negligible cost in terms of CPU time. Second, it provides a way for model comparison based on existing samples by specifying a virtual proposal distribution. This point can be explained more precisely as follows. More specifically, let $\pi_1(\mathbf{x}^{(1)}), \dots, \pi_m(\mathbf{x}^{(m)})$ denote the posterior distributions of m different models, $\mathcal{M}_1, \dots, \mathcal{M}_m$, respectively. Note that the m models may be of different dimensions. Let $\mathbf{x}_1^{(i)}, \dots, \mathbf{x}_{n_i}^{(i)}$ denote samples drawn from $\pi_i(\mathbf{x}^{(i)})$ for $i = 1, \dots, m$. The transition matrix

for different models can be estimated with these existing samples by specifying a virtual proposal distribution for jumping between models. The transition ratio can be calculated as in Eq. (18) based on the technique of resampling. The ratios of the normalizing constants of $\pi_i(\cdot)$'s can then be estimated by solving Eq. (6). This is beyond the ability of reversible jump MCMC (Green, 1995) and bridge sampling (Meng and Wong, 1996). The reversible jump MCMC method estimates the ratios of the normalizing constants of $\pi_i(\cdot)$'s only in a simulation process, while bridge sampling only works for the case that $\pi_i(\cdot)$'s are of the same dimension and have the same sample space.

Acknowledgements

The author thanks Professor Stanley P. Azen, the associate editor and two referees for their constructive comments and suggestions that have led to significant improvement of this article.

Appendix A.

Proof of Theorem 2.1. Let \widehat{P} denote either of the two estimators, $\widehat{P}^{(B)}$ and $\widehat{P}^{(M)}$. Let $\xi = (\pi(A_1), \dots, \pi(A_{m-1}))'$, \tilde{b} denote the underlying true vector of b (that is, $\tilde{b} = Eb$), $R = Q - bs' - I$, and $\tilde{R} = ER$. Thus, we have $\xi = -\tilde{R}^{-1}\tilde{b}$, and $\widehat{\xi} = -R^{-1}b$.

The ergodicity of the Markov chain implies that

$$\widehat{P} \rightarrow P \quad \text{almost surely} \tag{19}$$

as $n \rightarrow \infty$, and hence,

$$\begin{aligned} R &\rightarrow \tilde{R} \quad \text{almost surely,} \\ b &\rightarrow \tilde{b} \quad \text{almost surely,} \end{aligned} \tag{20}$$

as $n \rightarrow \infty$. For simplicity, we assume that

$$\begin{aligned} R &= \tilde{R} + \varepsilon cd', \\ b &= \tilde{b} + \varepsilon e, \end{aligned} \tag{21}$$

where c, d and e are all $(m - 1)$ -vectors of -1 's or $+1$'s, and $\varepsilon > 0$ denotes the size of the random error of \widehat{P}_{ij} . Then we have

$$\widehat{\xi} - \xi = -\varepsilon \left(\tilde{R}^{-1}e - \frac{\tilde{R}^{-1}cd'\xi}{1 + \varepsilon d'\tilde{R}^{-1}c} \right) + \frac{\varepsilon^2 \tilde{R}^{-1}cd'\tilde{R}^{-1}e}{1 + \varepsilon d'\tilde{R}^{-1}c}.$$

Thus,

$$\|\widehat{\xi} - \xi\| \sim O(\varepsilon). \tag{22}$$

Eqs. (20)–(22) imply that

$$\widehat{\xi} \rightarrow \xi, \quad \text{almost surely,}$$

as $n \rightarrow \infty$. \square

References

- Ashford, J.R., 1959. An approach to the analysis of data for semi-quantal responses in biological assay. *Biometrics* 15, 573–581.
- Barry, D., Hartigan, J.A., 1993. A Bayesian analysis for change point problems. *J. Amer. Statist. Assoc.* 88, 309–319.
- Casella, G., Roberts, C.P., 1996. Rao–Blackwellisation of sampling schemes. *Biometrika* 83, 81–94.
- Cobb, G.W., 1978. The problem of the Nile: conditional solution to a changepoint problem. *Biometrika* 65, 243–251.
- Freeman, J.M., 1986. An unknown change point and goodness of fit. *Statist.* 35, 335–344.
- Geyer, C.J., Thompson, E.A., 1995. Annealing Markov chain Monte Carlo with applications to pedigree analysis. *J. Amer. Statist. Assoc.* 90, 909–920.
- Green, P.J., 1995. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika* 82, 711–732.
- Hastings, W.K., 1970. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika* 57, 97–109.
- Liang, F., Wong, W.H., 2000. Evolutionary Monte Carlo sampling: applications to C_p model sampling and change-point problem. *Statist. Sinica* 10, 317–342.
- Marinari, E., Parisi, G., 1992. Simulated tempering: a new Monte Carlo scheme. *Europhys. Lett.* 19, 451–458.
- Meng, X.L., Wong, W.H., 1996. Simulating ratios of normalizing constants via a simple identity: a theoretical exploration. *Statist. Sinica* 6, 831–860.
- Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., Teller, E., 1953. Equation of state calculations by fast computing machines. *J. Chem. Phys.* 21, 1087–1091.
- Montgomery, D.C., peck, E.A., Vining, G.G., 2001. *Introduction to Linear Regression Analysis*, 3rd Edition. Wiley, New York.
- Phillips, D.B., Smith, A.F.M., 1996. Bayesian model comparison via jump diffusions. In: Gilks, W.R., Richardson, S., Spiegelhalter, D.J. (Eds.), *Markov Chain Monte Carlo in Practice*. Chapman & Hall, London, pp. 215–239.
- Roberts, G.O., 1996. Markov chain concepts related to sampling algorithms. In: Gilks, W.R., Richardson, S., Spiegelhalter, D.J. (Eds.), *Markov Chain Monte Carlo in Practice*. Chapman & Hall, London, pp. 45–57.
- Schulman, L.S., 1981. *Techniques and Applications of Path Integration*. Wiley, New York.
- Tierney, L., 1994. Markov chains for exploring posterior distributions (with Discussion). *Ann. Statist.* 22, 1701–1786.
- Winkler, G., 1995. *Image Analysis, Random Fields and Dynamic Monte Carlo Methods*. Springer, New York.