

Stochastic Approximation Monte Carlo and Its Applications

Faming Liang

Department of Statistics
Texas A&M University

-
1. Liang, F., Liu, C. and Carroll, R.J. (2007) Stochastic approximation in Monte Carlo computation. *JASA*, **102**, 305-320.
 2. Liang, F. (2007) Annealing stochastic approximation Monte Carlo for neural network training. *Machine Learning*, **68**, 201-233.
 3. Liang, F. (2007) Continuous contour Monte Carlo for marginal density estimation with an application to a spatial statistical model, *JCGS*, **16**(3), 608-632.
 4. Liang, F. (2007) Improving SAMC using smoothing methods: theory and applications. *Annals of Statistics*, to appear.
 5. Cheon, S. and Liang, F. (2008) Phylogenetic Tree Reconstruction Using Stochastic Approximation Monte Carlo. *BioSystems*, **91**, 94-107.
 6. Liang, F. (2007) Improving Stochastic Approximation Markov chain Monte Carlo by Trajectory Averaging. Submitted to *Bernoulli*.

-
7. Liang, F., Chen, M-H. and Joseph G. Ibrahim (2007) SAMC for Monte Carlo Integration with applications to high dimensional regression problems. Submitted to *Statistica Sinica*.

Two motivation examples

Example 1: Suppose we are interested in sampling from the following mixture Gaussian distribution,

$$f(x) = \frac{1}{3}N_2(\mu_1, \Sigma_1) + \frac{1}{3}N_2(\mu_2, \Sigma_2) + \frac{1}{3}N_2(\mu_3, \Sigma_3),$$

where

$$\mu_1 = \begin{pmatrix} -8 \\ -8 \end{pmatrix} \quad \Sigma_1 = \begin{pmatrix} 1 & 0.9 \\ 0.9 & 1 \end{pmatrix} \quad \mu_2 = \begin{pmatrix} 6 \\ 6 \end{pmatrix} \quad \Sigma_2 = \begin{pmatrix} 1 & -0.9 \\ -0.9 & 1 \end{pmatrix}$$

$$\mu_3 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad \Sigma_3 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

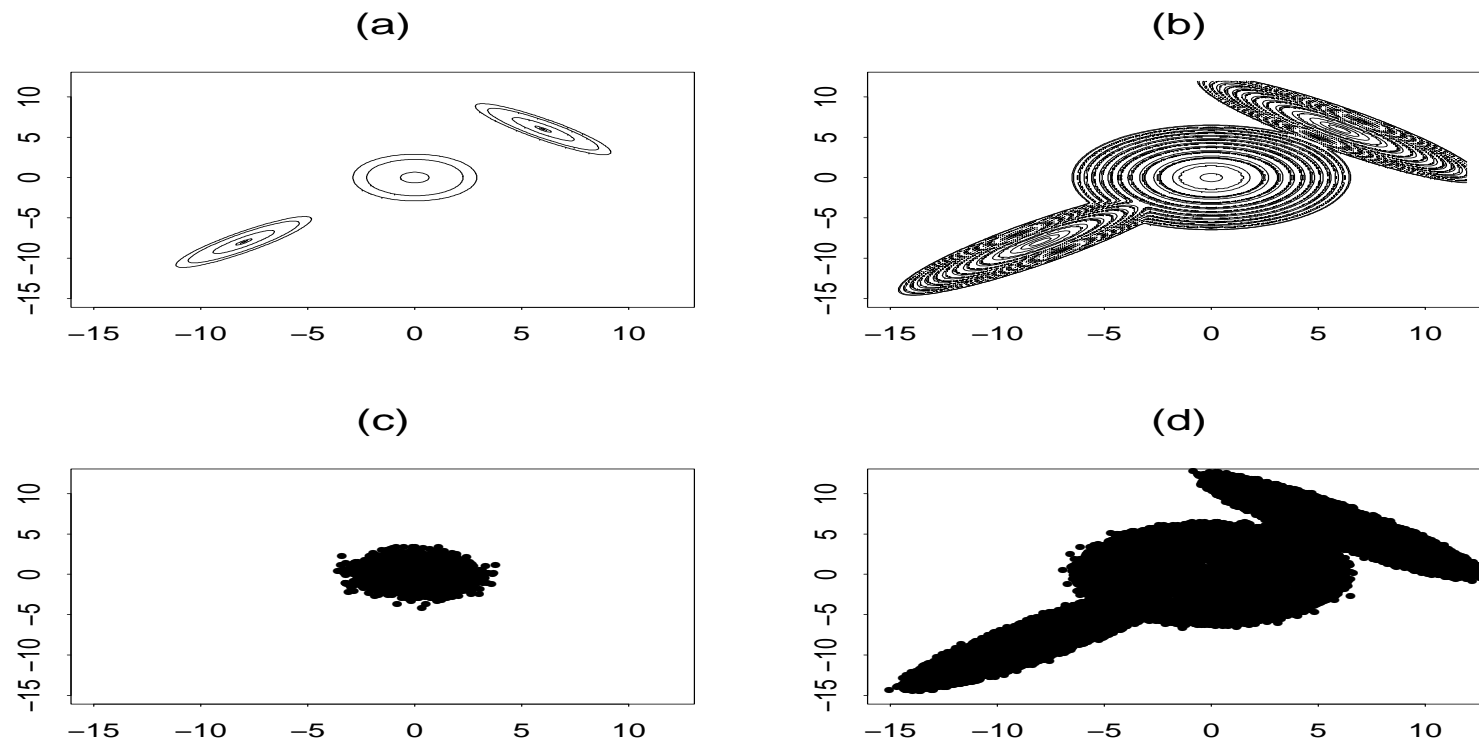


Figure 1: Plot (a) shows the contour plot of the density.

Example 2: Consider minimizing the following function on $[-1.1, 1.1]^2$

$$U(x, y) = -(x \sin(20y) + y \sin(20x))^2 \cosh(\sin(10x)x) \\ - (x \cos(10y) - y \sin(10x))^2 \cosh(\cos(20y)y),$$

whose global minimum is -8.12465, attained at $(x, y) = (-1.0445, -1.0084)$ and $(1.0445, -1.0084)$.

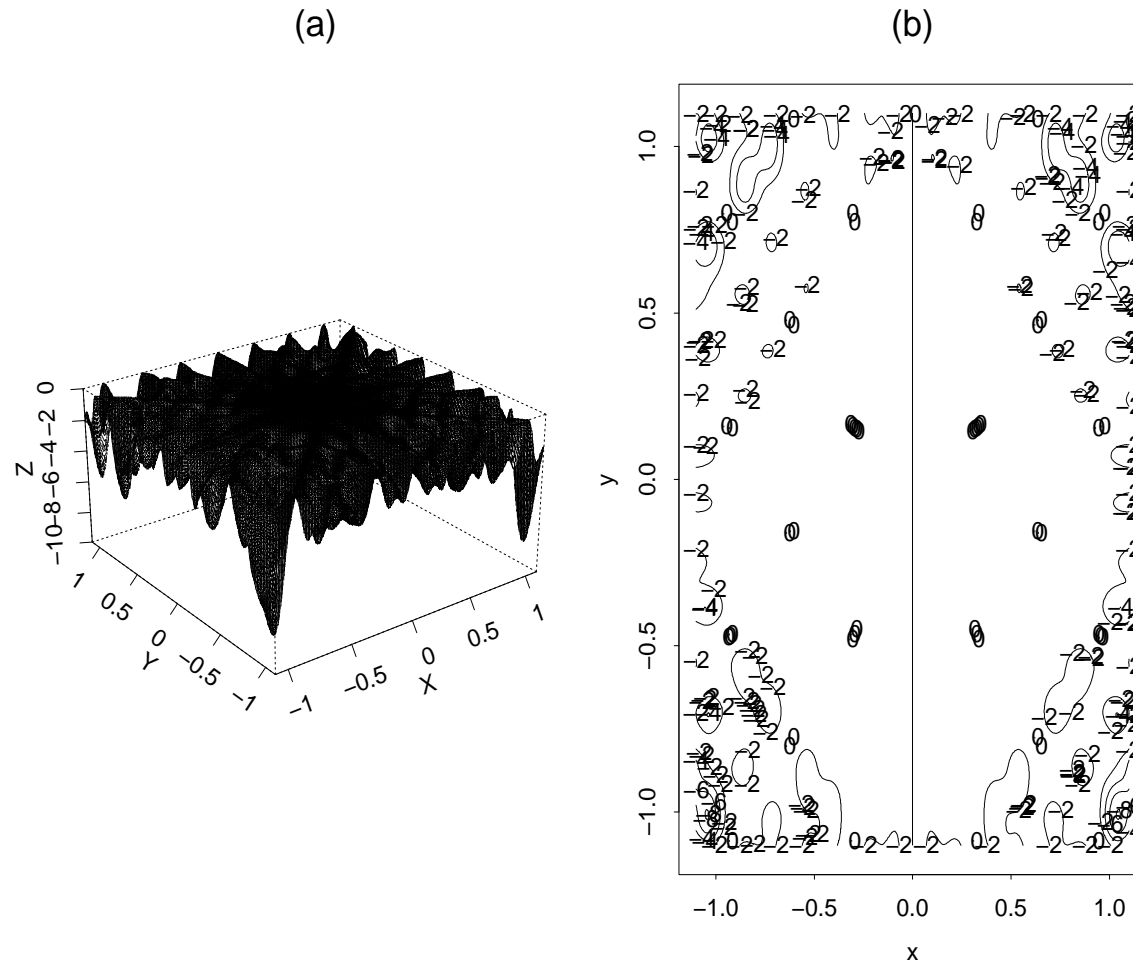


Figure 2: Grid and contour representation of a function defined on $[-1.1, 1.1]^2$.

The above examples can be formulated to simulate from a Boltzmann distribution,

$$f(x) = c\psi(x), \quad x \in \mathcal{X}, \quad (1)$$

where c is a constant, $\psi(x) = \exp(-U(x)/\tau)$, τ is called the temperature, and $U(x)$ is called the energy function.

Two basic MCMC algorithms

- (1) Metropolis-Hastings algorithm (Metropolis et al, 1953; Hastings, 1970)
- (2) The Gibbs sampler. (Geman and Geman, 1984).

Metropolis-Hastings Algorithm

- (a) Propose a new state y from a proposal distribution $T(x_t \rightarrow y)$, where x_t denotes the state of the Markov chain at time t .
- (b) Accept y with the probability

$$\min\left\{\frac{\psi(y)T(y \rightarrow x_t)}{\psi(x_t)T(x_t \rightarrow y)}, 1\right\}.$$

If it is accepted, set $x_{t+1} = y$, otherwise, set $x_{t+1} = x_t$.

Difficulty

On the energy landscape of these systems, there are a multitude of local minima separated by high energy barriers. The sampler tends to get trapped in one of local energy minima indefinitely, rendering the simulation ineffective.

Typical Problems in Scientific Computation

1. Protein folding.
2. Phylogenetic tree reconstruction.
3. Neural Networks.
4. Some spatial statistical problems, e.g., Ising model, disease mapping.

Strategies for improving MCMC

1. The use of auxiliary variables:

- Swendsen-Wang algorithm (Swendsen and Wang, 1987)
- Parallel tempering (Geyer, 1991)
- Simulated tempering (Marinari and Parisi, 1992)
- Evolutionary Monte Carlo (Liang and Wong, 2001)

Strength and weakness: The temperature is typically treated as an auxiliary variable. Simulations at high temperatures broaden the space of sampling, and thus are able to help the system to escape from local energy minima.

2. The use of past samples:

- Multicanonical (Berg and Neuhaus, 1991)
- $1/k$ -ensemble algorithm (Hesselbo and Stinchcombe, 1995; Liang, 2004)
- Wang-Landau (WL) Algorithm (Wang and Landau, 2001; Liang, 2005)
- Dynamic weighting (Wong and Liang, 1997)
- Dynamically weighted importance sampling (Liang, 2002)

Strength and weakness:

- **Dynamic weighting**: The variability of the weights is too high.
- **Multicanonical and related algorithms**: They are usually used for discrete systems. In the multicanonical algorithm, the trial distribution is defined as:

$$f^*(x) = \frac{1}{\#\{y : U(y) = U(x)\}},$$

where x and y take values on a discrete set.

There is no rigorous theory to support their convergence.

Basic Idea

- Partition the sample space into different subregions: E_1, \dots, E_m , $\bigcup_{i=1}^m E_i = \mathcal{X}$, and $E_i \cap E_j = \emptyset$ for $i \neq j$.
- Let $g_i = \int_{E_i} \psi(x) dx$, and choose $\pi = (\pi_1, \dots, \pi_m)$, $\pi_i \geq 0$, and $\sum_i \pi_i = 1$.
- Sampling from the distribution

$$p_\theta(x) \propto \sum_{i=1}^m \frac{\psi(x)}{e^{\theta^{(i)}}} I(x \in E_i).$$

If $\theta^{(i)} = \log(g_i/\pi_i)$ for all i , sampling from $p_\theta(x)$ will result in a random walk in the space of subregions with each subregion being sampled with probability π_i (viewing each subregion as a “single point”). Therefore, sampling from $p_\theta(x)$ can avoid the local trap problem encountered in sampling from $f(x)$.

Algorithm Setting

- *Condition* (A_1) The sequence $\{a_k\}_{k=0}^{\infty}$ is non-increasing, positive and satisfies the conditions:

$$\sum_{k=1}^{\infty} a_k = \infty, \quad \lim_{k \rightarrow \infty} (ka_k) = \infty, \quad \lim_{k \rightarrow \infty} (a_{k+1}^{-1} - a_k^{-1}) = 0, \quad (2)$$

and for some $\tau \in (0, 1)$

$$\sum_{k=1}^{\infty} \frac{a_k^{(1+\tau)/2}}{\sqrt{k}} < \infty \quad (3)$$

It is clear that $a_k = 1/k^\eta$, $\forall \eta \in (1/2, 1]$, satisfies (2). Then (3) holds for any $\tau > 1/\eta - 1$.

Algorithm

1. (Sampling) Draw sample x_{k+1} with a single MH iteration for which the invariant distribution is

$$\hat{p}_k(x) = \frac{1}{Z_k} \sum_{i=1}^m \frac{\psi(x)}{e^{\theta_k^{(i)}}} I(x \in E_i).$$

2. (Weight updating) Set

$$\theta_{k+1}^* = \theta_k + a_{k+1} H(\theta_k, x_{k+1}),$$

where $H(\theta_k, x_{k+1}) = \mathbf{e}_{x_{k+1}} - \pi$ and $\mathbf{e}_{x_{k+1}} = (I(x_{k+1} \in E_1), \dots, I(x_{k+1} \in E_m))$.

3. (Varying truncation) If $\theta_{k+1}^* \in \Theta$, set $\theta_{k+1} = \theta_{k+1}^*$. Otherwise, set $\theta_{k+1} = \theta_{k+1}^* + c^*$, where c^* is chosen such that $\theta_{k+1}^* + c^* \in \Theta$.

Lyapunov condition on $h(\theta)$

Let $\langle x, y \rangle$ denote the Euclidean inner product.

(A₂) The function $h : \Theta \rightarrow \mathbb{R}^d$ is continuous, and there exists a continuously differentiable function $v : \Theta \rightarrow [0, \infty)$ such that

(i) For any integer $M > 0$, the level set $\mathcal{V}_M = \{\theta \in \Theta, v(\theta) \leq M\} \subset \Theta$ is compact.

(ii) There exists $M_0 > 0$ such that

$$\tilde{\Theta} = \{\theta \in \Theta, \langle \nabla v(\theta), h(\theta) \rangle = 0\} \subset \text{int}(\mathcal{V}_{M_0}),$$

and $\langle \nabla v(\theta), h(\theta) \rangle < 0$ for any $\theta \in \Theta \setminus \mathcal{V}_{M_0}$, where $\text{int}(A)$ denotes the interior of set A .

(iii) For all $\theta \in \Theta$, $\langle \nabla v(\theta), h(\theta) \rangle \leq 0$, and $\text{int}(v(\tilde{\Theta})) = \emptyset$.

Stability condition on $h(\theta)$

(A_3) *The mean field function $h(\theta)$ is measurable and locally bounded. There exist a stable matrix F (i.e., all eigenvalues of F are with negative real parts), $\gamma > 0$, and $\rho \in (\tau, 1]$ such that for any point $\theta^0 \in \tilde{\Theta}$,*

$$\|h(\theta) - F(\theta - \theta^0)\| \leq c_1 \|\theta - \theta^0\|^{1+\rho}, \quad \forall \theta \in \{\theta : \|\theta - \theta^0\| \leq \gamma\},$$

where c_1 is a constant.

Drift condition

(A_4) For any $\theta \in \Theta$, the transition kernel P_θ is irreducible and aperiodic. In addition, there exists a function $V : \mathcal{X}^k \rightarrow [1, \infty)$ and constants $\alpha \geq 2$ and $\beta \in (0, 1]$ such that for any compact subset $\mathcal{K} \subset \Theta$,

(i) There exist a set $\mathbf{C} \subset \mathcal{X}$, an integer l , constants $0 < \lambda < 1$, $b, \varsigma, \delta > 0$ and a probability measure ν such that

$$\bullet \quad \sup_{\theta \in \mathcal{K}} P_\theta^l V^\alpha(x) \leq \lambda V^\alpha(x) + bI(x \in \mathbf{C}), \quad \forall x \in \mathcal{X} \quad (4)$$

$$\bullet \quad \sup_{\theta \in \mathcal{K}} P_\theta V^\alpha(x) \leq \varsigma V^\alpha(x), \quad \forall x \in \mathcal{X}. \quad (5)$$

$$\bullet \quad \sup_{\theta \in \mathcal{K}} P_\theta^l(x, A) \geq \delta \nu(A), \quad \forall x \in \mathbf{C}, \forall A \in \mathcal{B}. \quad (6)$$

(ii) *There exists a constant c such that for all $x \in \mathcal{X}$,*

- $\sup_{\theta \in \mathcal{K}} \|H(\theta, x)\| \leq cV(x). \quad (7)$
- $\sup_{(\theta, \theta') \in \mathcal{K}} \|H(\theta, x) - H(\theta', x)\| \leq cV(x)\|\theta - \theta'\|^\beta \quad (8)$

(iii) *There exists a constant c such that for all $(\theta, \theta') \in \mathcal{K} \times \mathcal{K}$,*

- $\|P_\theta g - P_{\theta'} g\|_V \leq c_2 \|g\|_V |\theta - \theta'|^\beta, \quad \forall g \in \mathcal{L}_V. \quad (9)$
- $\|P_\theta g - P_{\theta'} g\|_{V^\alpha} \leq c_2 \|g\|_{V^\alpha} |\theta - \theta'|^\beta, \quad \forall g \in \mathcal{L}(V^\alpha) \quad (10)$

Theoretical Results

Lemma 1 *Assume the drift condition (A_4) and $\sup_{x \in \mathcal{X}} V(x) < \infty$. Let $\epsilon_k = H(\theta_k, x_{k+1}) - h(\theta_k)$. There exist \mathbb{R}^d -valued random processes $\{e_k\}_{k \geq 1}$, $\{\nu_k\}_{k \geq 1}$, and $\{s_k\}_{k \geq 1}$ defined on a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ such that*

(i) $\epsilon_k = e_k + \nu_k + s_k$.

(ii) $\{e_k\}$ is a martingale difference sequence, and $\frac{1}{\sqrt{n}} \sum_{k=1}^n e_k \longrightarrow N(0, Q)$ in distribution, where $Q = \lim_{k \rightarrow \infty} E(e_k e_k')$.

(iii) $E\|\nu_k\| = O(a_k^{(1+\tau)/2})$, where τ is given in condition (A_1) .

(iv) $\left\| \sum_{k=0}^n a_k s_k \right\| = O(a_n)$.

THEOREM 1 (*Convergence*) (Liang et al., 2007)

Let α_σ denote the number of iterations for which the σ -th truncation occurs in the SAMC simulation. Assume the conditions (A_1) and (A_2) hold, and there exists a drift function $V(x)$ such that $\sup_{x \in \mathcal{X}} V(x) < \infty$ and the drift condition (A_4) holds. Then there exists a number σ such that $\alpha_\sigma < \infty$ a.s., $\alpha_{\sigma+1} = \infty$ a.s., and $\{\theta_k\}$ given by the SAMC algorithm has no truncation for $k \geq \alpha_\sigma$, i.e.,

$$\theta_{k+1} = \theta_k + a_k H(\theta_k, x_{k+1}), \quad \forall k \geq \alpha_\sigma,$$

and

$$\theta_k^{(i)} \rightarrow \begin{cases} c + \log(\int_{E_i} \psi(x) dx) - \log(\pi_i + \nu), & \text{if } E_i \neq \emptyset, \\ -\infty. & \text{if } E_i = \emptyset, \end{cases} \quad (11)$$

where $\nu = \sum_{j \in \{i: E_i = \emptyset\}} \pi_j / (m - m_0)$ and m_0 is the number of empty subregions. The constant c can be determined by imposing an extra constraint on θ_k , e.g., $\theta_k^{(m)} = 0$ for all $k \geq 0$.

THEOREM 2 (*Averaging Normality*) (Liang 2007, submitted)

Under the conditions of Theorem 2, we have

$$\bar{\theta}_k = \frac{1}{k} \sum_{i=1}^k \theta_i$$

is asymptotically efficient; that is,

$$\sqrt{k}(\bar{\theta}_k - \theta^0) \longrightarrow N(\mathbf{0}, S) \quad \text{as } k \rightarrow \infty,$$

where $S = F^{-1}Q(F^{-1})^T$, *and* Q *is as defined in Lemma 1.*

THEOREM 3 (*IWIW property*) (Liang et al., 2007, submitted)

If the desired sampling distribution is uniform over all the subregions, i.e., $\pi_1 = \dots = \pi_m = 1/m$ SAMC is invariant with respect to the importance weights (IWIW).

This Theorem implies that the integral $E_f h(x)$ can be estimated on-line by

$$\widehat{E_f h(x)} = \frac{\sum_{k=1}^n w_k h(x_k)}{\sum_{k=1}^n w_k},$$

where $w_k = \sum_{i=1}^m e^{\theta_k^{(i)}} I(x_k \in E_i)$. As $n \rightarrow \infty$,

$$\widehat{E_f h(x)} \rightarrow E_f h(x),$$

for the same reason that the usual importance sampling estimate converges.

Implementation Issues

1. **Sample space partition.** *It can be made according to our goal and the complexity of the problem. Here are some examples:*

(a) *Importance sampling: Energy function, maximum energy difference ≤ 2 .*

(b) *Model selection: Model index.*

2. **Desired sampling distribution.**

(a) *Set π to bias the sampling to low energy regions if we aim to minimize the energy function.*

(b) *Set π to be uniform if we aim at estimation.*

3. **Choices of η , t_0 and the number of iterations.** *The diagnostic statistic:*

$$\epsilon_f(E_i) = \begin{cases} \frac{\hat{\pi}_i - (\pi_i + \nu)}{\pi_i + \nu} \times 100\%, & \text{if } E_i \neq \emptyset, \\ 0, & \text{if } E_i = \emptyset, \end{cases} \quad (12)$$

for $i = 1, \dots, m$. If $\max_{i=1}^m |\epsilon_f(E_i)|$ is large, say, greater than 10%, the convergence of the run should be questioned. In this case, SAMC should be re-run with more iterations, a larger value of t_0 , or a smaller value of η .

x	1	2	3	4	5	6	7	8	9	10
$f(x)$	1	100	2	1	3	3	1	200	2	1

Table 1: The unnormalized mass function of the 10-state distribution.

Table 2: Comparison of SAMC and MH for the 10-state example, where the Bias and Standard Error (of the Bias) were calculated based on 100 independent runs.

Algorithm	Bias ($\times 10^{-3}$)	Standard Error ($\times 10^{-3}$)	CPU time (seconds)
SAMC	-0.528	1.513	0.38
MH	-3.685	4.634	0.20

The sample space was partitioned according to the mass function into five subregions: $E_1 = \{8\}$, $E_2 = \{2\}$, $E_3 = \{5, 6\}$, $E_4 = \{3, 9\}$ and $E_5 = \{1, 4, 7, 10\}$. The desired sampling distribution is uniform over 5 subregions.

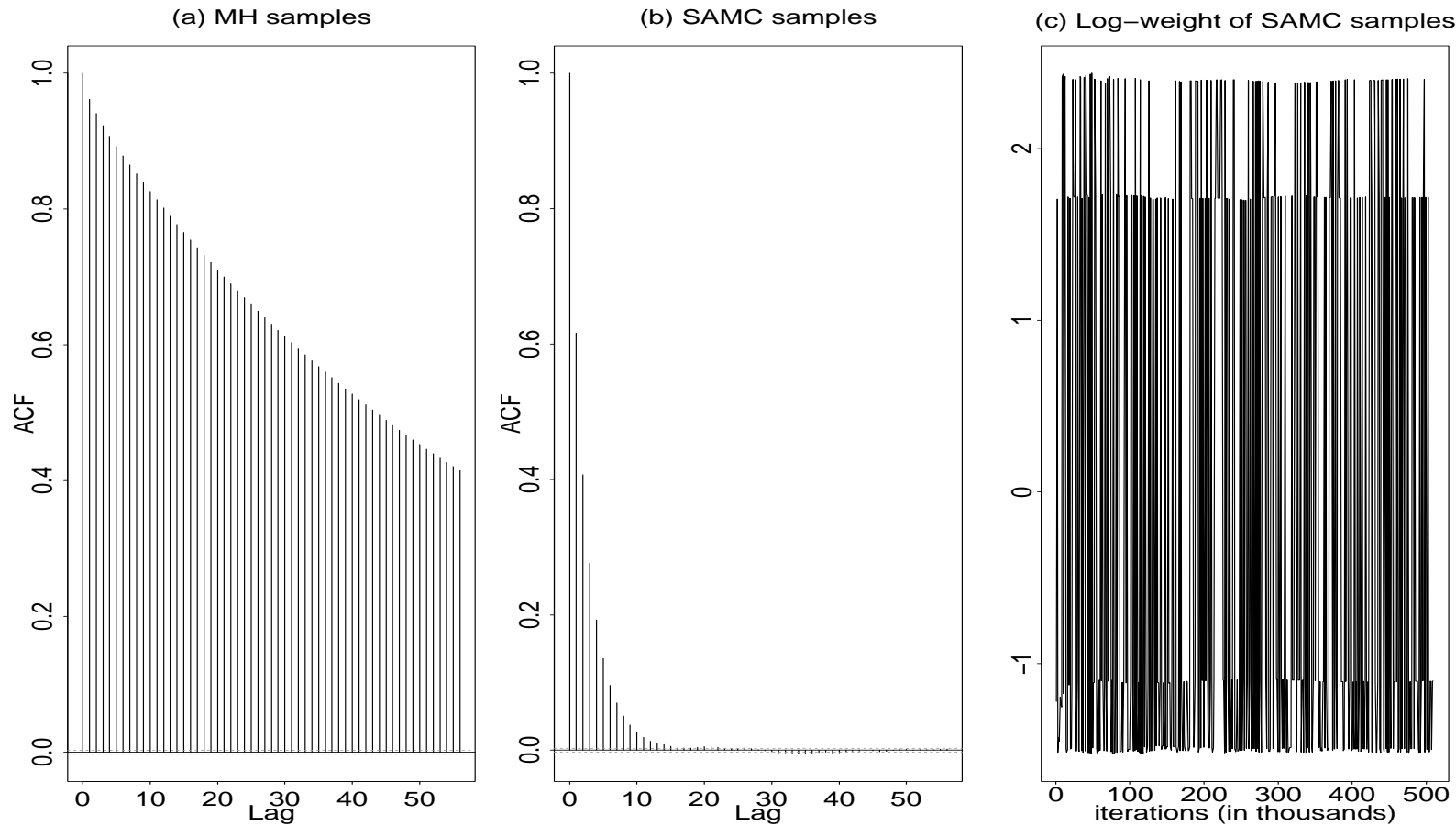


Figure 3: Computational results for the 10-state example. (a) Autocorrelation plot of the MH samples. (b) Autocorrelation plot of the SAMC samples. (c) Log-weight of the SAMC samples.

Let $\mathbf{s} = \{s_i : i \in D\}$ denote the observed binary data, where s_i is called a spin and D is the set of indices of the spins. Let $|D|$ denote the total number of spins in D , and $N(i)$ denote a set of “neighbors” of spin i . The likelihood function of the model is

$$f(\mathbf{s}|\alpha, \beta) = \frac{1}{\varphi(\alpha, \beta)} \exp \left\{ \alpha \sum_{i \in D} s_i + \frac{\beta}{2} \sum_{i \in D} s_i \left(\sum_{j \in N(i)} s_j \right) \right\}, \quad (13)$$

where $(\alpha, \beta) \in \Omega$, and

$$\varphi(\alpha, \beta) = \sum_{\text{for all possible } \mathbf{s}} \exp \left\{ \alpha \sum_{j \in D} s_j + \frac{\beta}{2} \sum_{i \in D} s_i \left(\sum_{j \in N(i)} s_j \right) \right\}.$$

When β is large, say, 0.5, the configuration \mathbf{s} tends to have large clusters of the same orientation, which fluctuate very slowly.

Methods to resolve the difficulty in normalizing constant evaluation:

(a) Working on a pseudo-likelihood function (Besag, 1975):

$$PL(\alpha, \beta | \mathbf{s}) = \prod_{i \in D} \frac{e^{s_i(\alpha + \beta \sum_{j \in N(i)} s_j)}}{e^{\alpha + \beta \sum_{j \in N(i)} s_j} + e^{-\alpha - \beta \sum_{j \in N(i)} s_j}}. \quad (14)$$

The resulting estimate is called MPLE.

(b) Working on a Monte Carlo log-likelihood (up to a constant)(Geyer and Thompson, 1992):

$$L_n(\alpha, \beta | \mathbf{s}) = \alpha \sum_{i \in D} s_i + \frac{\beta}{2} \sum_{i \in D} s_i \left(\sum_{j \in N(i)} s_j \right) - \log \left[\frac{1}{n} \sum_{k=1}^n \frac{\psi(\alpha, \beta, \mathbf{s}^{(k)})}{\psi(\alpha^*, \beta^*, \mathbf{s}^{(k)})} \right]. \quad (15)$$

The resulting estimate is called MCMLE.

A natural choice for the trial distribution is a mixture distribution of the form

$$p_{mix}^*(\mathbf{s}) = \frac{1}{m^*} \sum_{j=1}^{m^*} p(\mathbf{s} | \alpha_j, \beta_j), \quad (16)$$

where the values of the parameters $(\alpha_1, \beta_1), \dots, (\alpha_{m^}, \beta_{m^*})$ are pre-specified. To complete this idea, the key is to estimate $\varphi(\alpha_j, \beta_j), \dots, \varphi(\alpha_{m^*}, \beta_{m^*})$ (up to a common multiplicative constant).*

Estimate	single-MCMLE	SAMC
$\text{RMSE}(T_1^{sim})$	59.51	2.90
$\text{RMSE}(T_2^{sim})$	114.91	4.61

Table 3: Comparison of the accuracy of the SAMC and single-MCMLEs for the US cancer data. $T_1 = \sum_i s_i$, $T_2 = \sum_i s_i (\sum_j s_j) / 2$, $\text{RMSE}(T_i^{sim})$ is calculated as $\sqrt{\sum_{k=1}^5 (T_i^{sim,k} - T_i^{obs})^2 / 5}$, where $i = 1, 2$, and $T_i^{sim,k}$ denotes the value of T_i^{sim} calculated based on the k^{th} estimate of (α, β) .

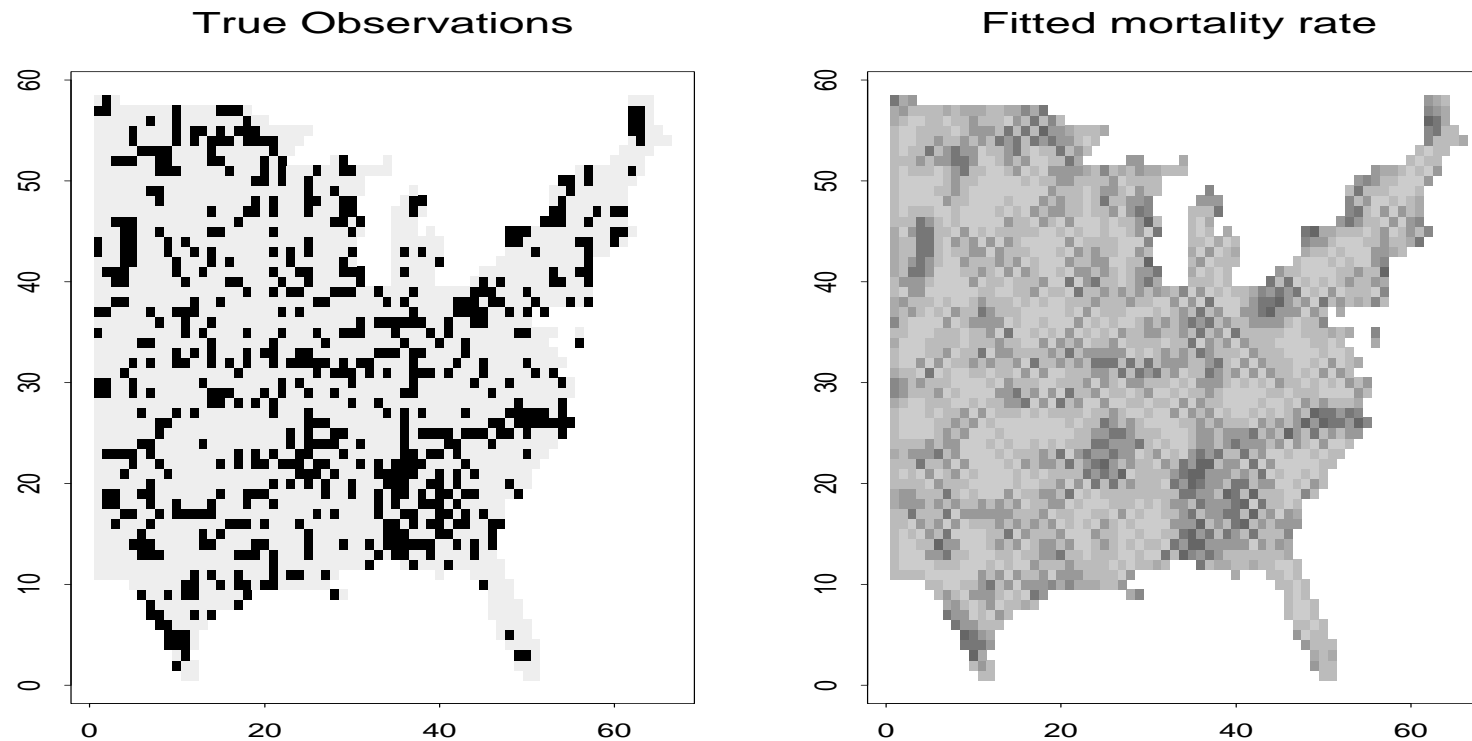


Figure 4: The U.S. cancer mortality rate data. (a) The mortality map of liver and gallbladder cancer (including bile ducts) for white males during the decade 1950-1959. The black squares denote the counties of high cancer mortality rate, and the white squares denote the counties of low cancer mortality rate. (b) Fitted cancer mortality rates. The cancer mortality rate of each county is represented by the gray level of the corresponding square.

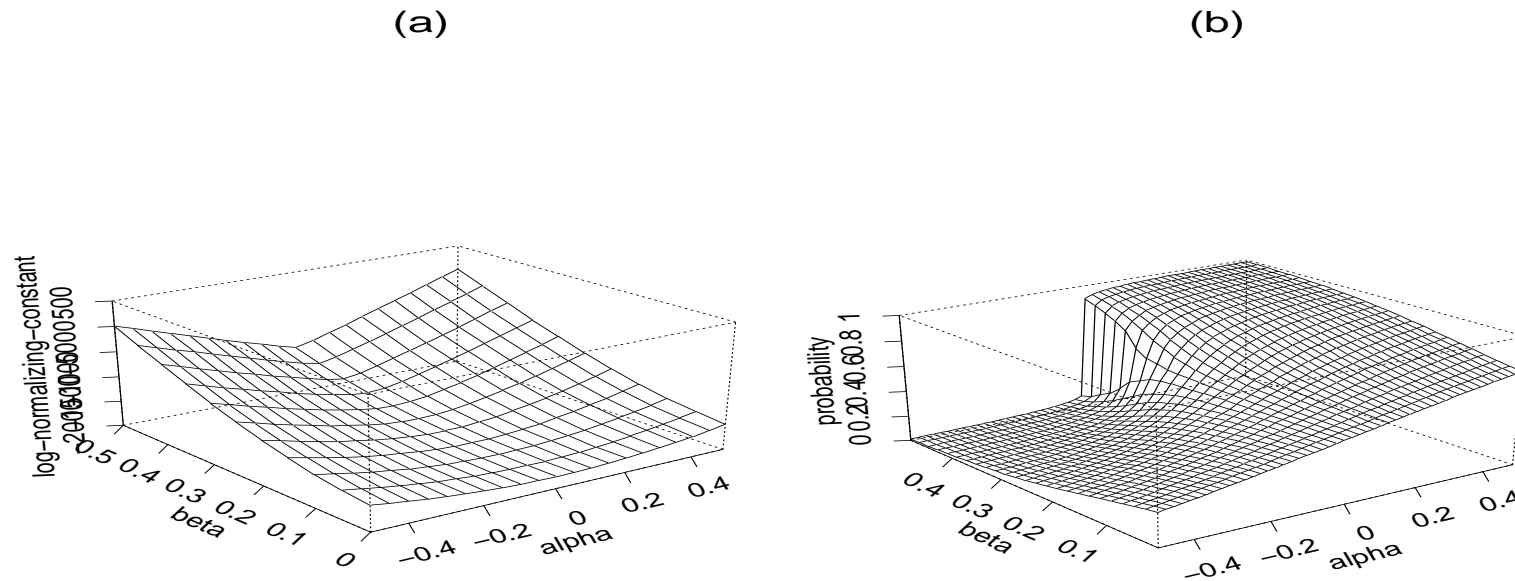


Figure 5: Computational results of SAMC. (a) Estimate of $\log \varphi(\alpha, \beta)$ on a 21×11 lattice with $\alpha \in \{-0.5, -0.45, \dots, 0.5\}$ and $\beta \in \{0, 0.05, \dots, 0.5\}$. (b) Estimate of $P(s_i = +1 | \alpha, \beta)$ on a 50×25 lattice with $\alpha \in \{-0.49, -0.47, \dots, 0.49\}$ and $\beta \in \{0.01, 0.03, \dots, 0.49\}$.

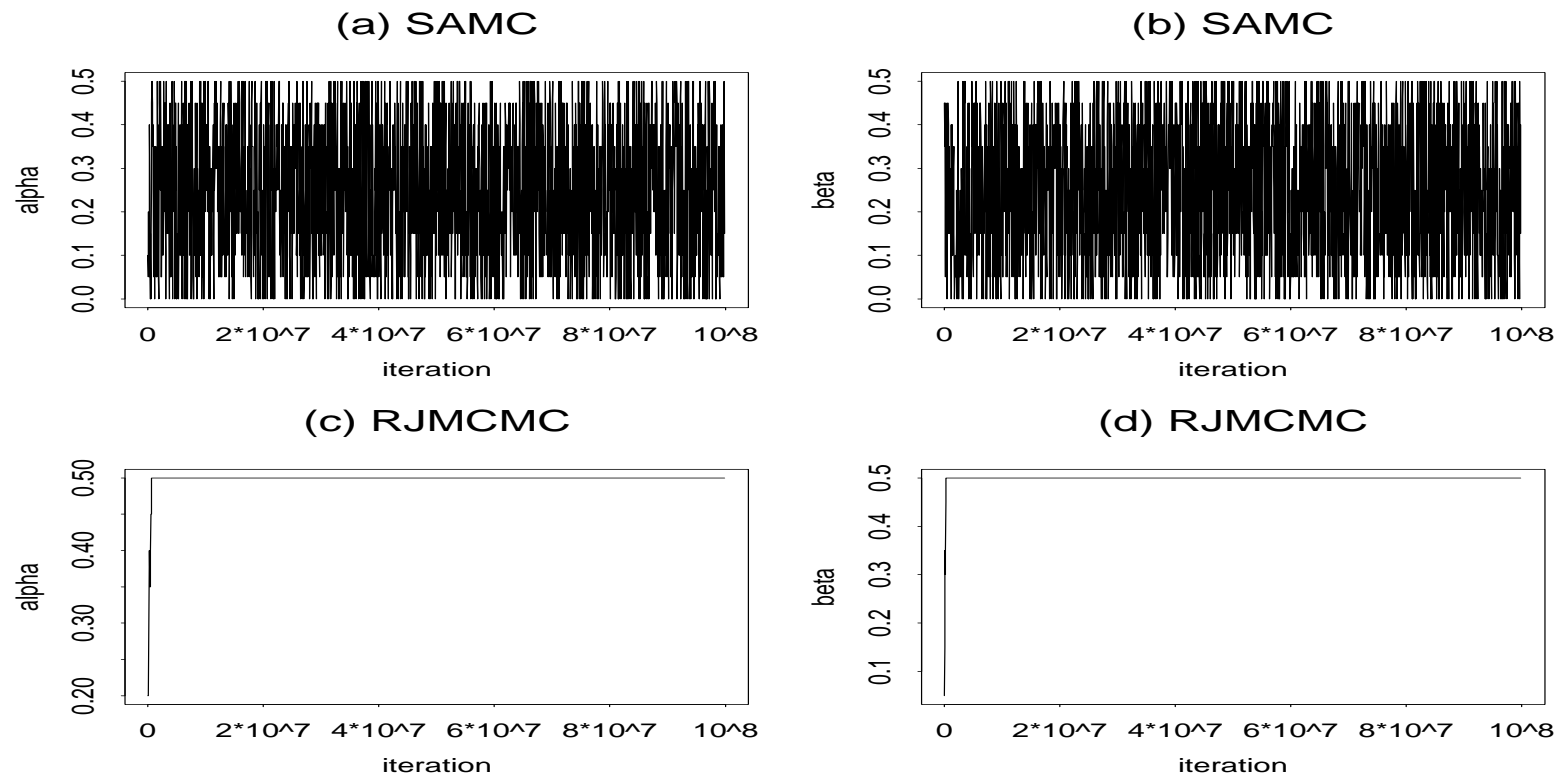


Figure 6: Comparison of SAMC and RJMCMC. Plots (a) and (b) show, respectively, the sample paths of α and β in a run of SAMC. Plots (c) and (d) show, respectively, the sample paths of α and β in a run of RJMCMC.

Motivation for Smoothing SAMC

Intuitively, x_t may contain some information on its neighboring subregions, so the visiting to its neighboring subregions should also be penalized to some extent in the next iteration.

The efficiency of SAMC can be improved by including at each iteration a smoothing step, which distributes the information contained in each sample to its neighboring subregions. The new algorithm is thus called smoothing-SAMC or SSAMC for simplicity.

Motivation Examples

We note that for many problems, E_1, \dots, E_m can be regarded as a sequence of naturally ordered categories. Here are some examples.

- *Model selection: The model space \mathcal{X} can be partitioned according to the index of models, and the subregions can be naturally ordered according to the number of parameters contained in each model.*
- *Function optimization: The solution space \mathcal{X} can be partitioned according to the objective function, and the subregions can also be naturally ordered according to the objective function.*

Suppose that $x_k^{(1)}, \dots, x_k^{(\kappa)}$ are samples generated using a MH kernel with the invariant distribution $p_{\theta_k}(x)$. Since κ is usually a small number, say, 10 to 20, the samples form a sparse frequency vector $\mathbf{e}_{x_k} = (e_k^{(i)}, \dots, e_k^{(m)})$ with $e_k^{(i)} = \sum_{l=1}^{\kappa} I(x_k^{(l)} \in E_i)$.

The frequency estimate can be improved by a smoothing method. The Nadaraya-Watson (NW) kernel estimator works as follows:

$$\hat{p}_k^{(i)} = \frac{\sum_{j=1}^m W\left(\frac{\Lambda(i-j)}{mh_k}\right) \frac{e_k^{(j)}}{\kappa}}{\sum_{j=1}^m W\left(\frac{\Lambda(i-j)}{mh_k}\right)}, \quad (17)$$

where $W(z)$ is a kernel function with bandwidth h_k , and Λ is a rough estimate of the range of $\lambda(x)$, $x \in \mathcal{X}$.

By assuming that $W(z)$ has a bounded support, we can show

$$\hat{p}_k^{(i)} - e_k^{(i)} / \kappa = O(h_k).$$

- (a) **(Sampling)** Simulate samples $x_k^{(1)}, \dots, x_k^{(\kappa)}$ using the MH algorithm with the proposal distribution $q(x_{k(i)}, \cdot)$ and the invariant distribution $p_{\theta_k}(x)$, where $x_k^{(0)} = x_{k-1}^{(\kappa)}$.
- (b) **(Smoothing)** Calculate $\hat{p}_k = (\hat{p}_k^{(i)}, \dots, \hat{p}_k^{(m)})$ using a kernel smoothing method.
- (c) **(Weight updating)** Set

$$\theta^* = \theta_k + a_{k+1}(\hat{p}_k - \pi). \quad (18)$$

If $\theta^* \in \Theta$, set $\theta_{k+1} = \theta^*$; otherwise, set $\theta_{k+1} = \theta^* + c^*$, where c^* can be any number which satisfies the condition $\theta^* + c^* \in \Theta$.

Notations

- Let $Z = (z_1, z_2, \dots, z_n)$ denote a sequence of independent observations.
- Let $\vartheta^{(k)}$ denote a configuration of ϑ with k ones, which represents a model of k change points.
- Let $\eta^{(k)} = (\vartheta^{(k)}, \mu_1, \sigma_1^2, \dots, \mu_{k+1}, \sigma_{k+1}^2)$.
- Let \mathcal{X}_k denote the space of models with k change points, $\vartheta^{(k)} \in \mathcal{X}_k$, and $\mathcal{X} = \cup_{k=0}^n \mathcal{X}_k$.

Assuming appropriate prior distributions, integrating out the parameters $\mu_1, \sigma_1^2, \dots, \mu_{k+1}, \sigma_{k+1}^2$ from the full posterior distribution, and taking a logarithm, we have

$$\begin{aligned}
& \log P(\boldsymbol{\vartheta}^{(k)} | Z) \\
&= a_k + \frac{k+1}{2} \log 2\pi - \sum_{i=1}^{k+1} \left\{ \frac{1}{2} \log(c_i - c_{i-1}) - \log \Gamma\left(\frac{c_i - c_{i-1} - 1}{2} + \alpha\right) \right. \\
&+ \left. \left(\frac{c_i - c_{i-1} - 1}{2} + \alpha\right) \log \left[\beta + \frac{1}{2} \sum_{j=c_{i-1}+1}^{c_i} z_j^2 - \frac{\left(\sum_{j=c_{i-1}+1}^{c_i} z_j\right)^2}{2(c_i - c_{i-1})} \right] \right\}.
\end{aligned} \tag{19}$$

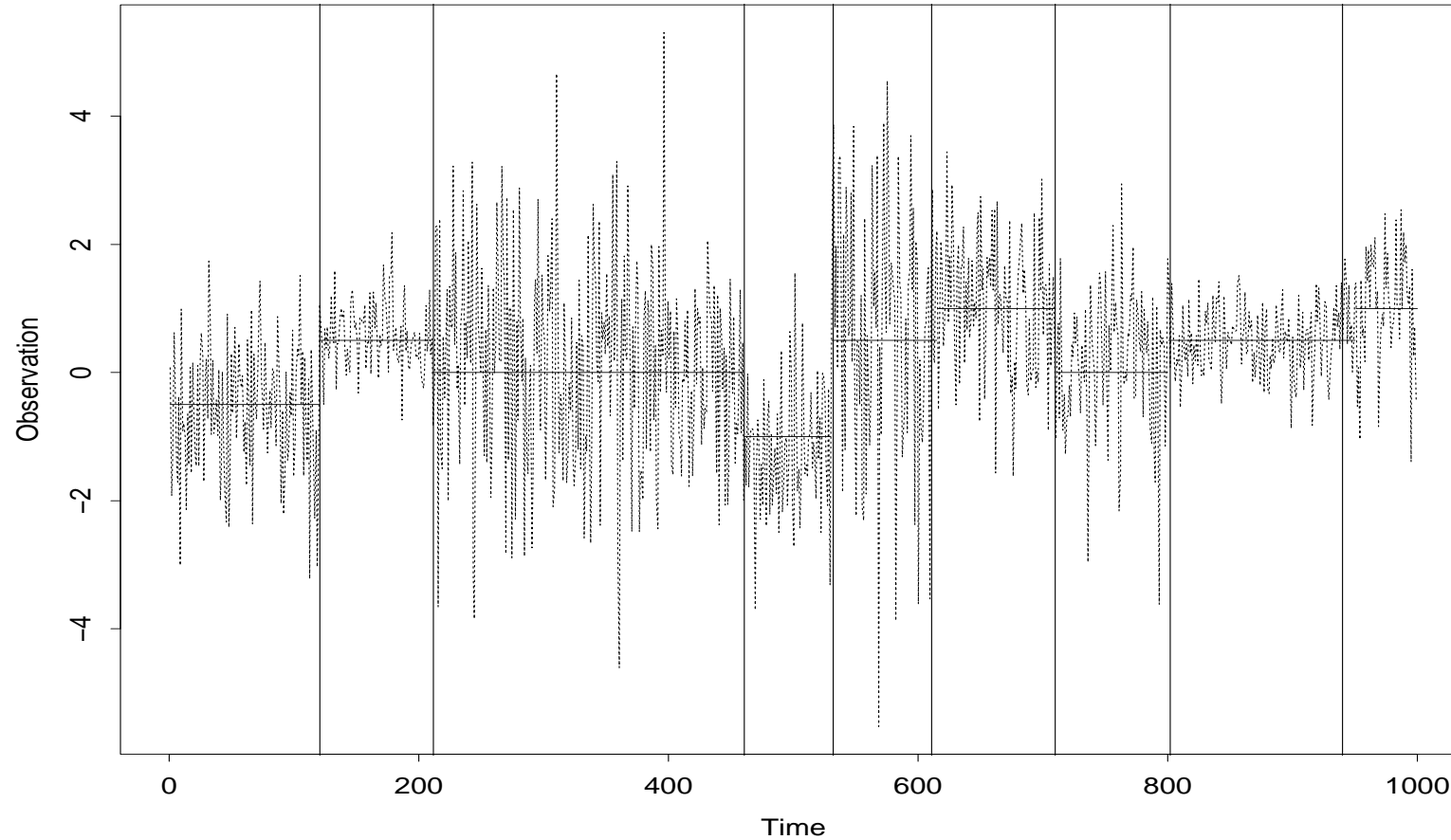


Figure 7: Comparison of the true change-point pattern (horizontal lines) and its MAP estimate (vertical lines).

k	SSAMC		SAMC		RJMCMC	
	prob(%)	SD	prob(%)	SD	prob(%)	SD
7	0.1010	0.0023	0.0944	0.0029	0.0907	0.0046
8	55.4666	0.2470	55.3928	0.6112	55.5726	0.3451
9	33.3744	0.1659	33.3728	0.3573	33.2117	0.2052
10	9.2982	0.1026	9.3647	0.2788	9.3537	0.1441
11	1.5655	0.0287	1.5785	0.0685	1.5694	0.0400
12	0.1768	0.0042	0.1803	0.0097	0.1845	0.0097
13	0.0157	0.0005	0.0154	0.0009	0.0165	0.0011
14	0.0018	0.0001	0.0011	0.0001	0.0009	0.0002

Table 4: The estimated posterior distribution $P(\mathcal{X}_k|Z)$ for the change-point identification example. SD: standard deviation of the estimates.

Algorithm	Mean	Standard Error	Minimum	Maximum	Proportion
SAMC	-8.12369	0.00015	-8.12465	-8.11244	99
Annealing-1	-8.07721	0.01483	-8.12465	-6.92783	31
Annealing-2	-8.06147	0.02066	-8.12465	-6.64358	22
Annealing-3	-7.54252	0.10004	-8.12465	-5.51122	28

Table 5: Comparison of SAMC and simulated annealing. Annealing-1, Annealing-2, and Annealing-3 correspond to the runs with $t_{high} = 5$, $t_{high} = 2$, and $t_{high} = 1$, respectively.

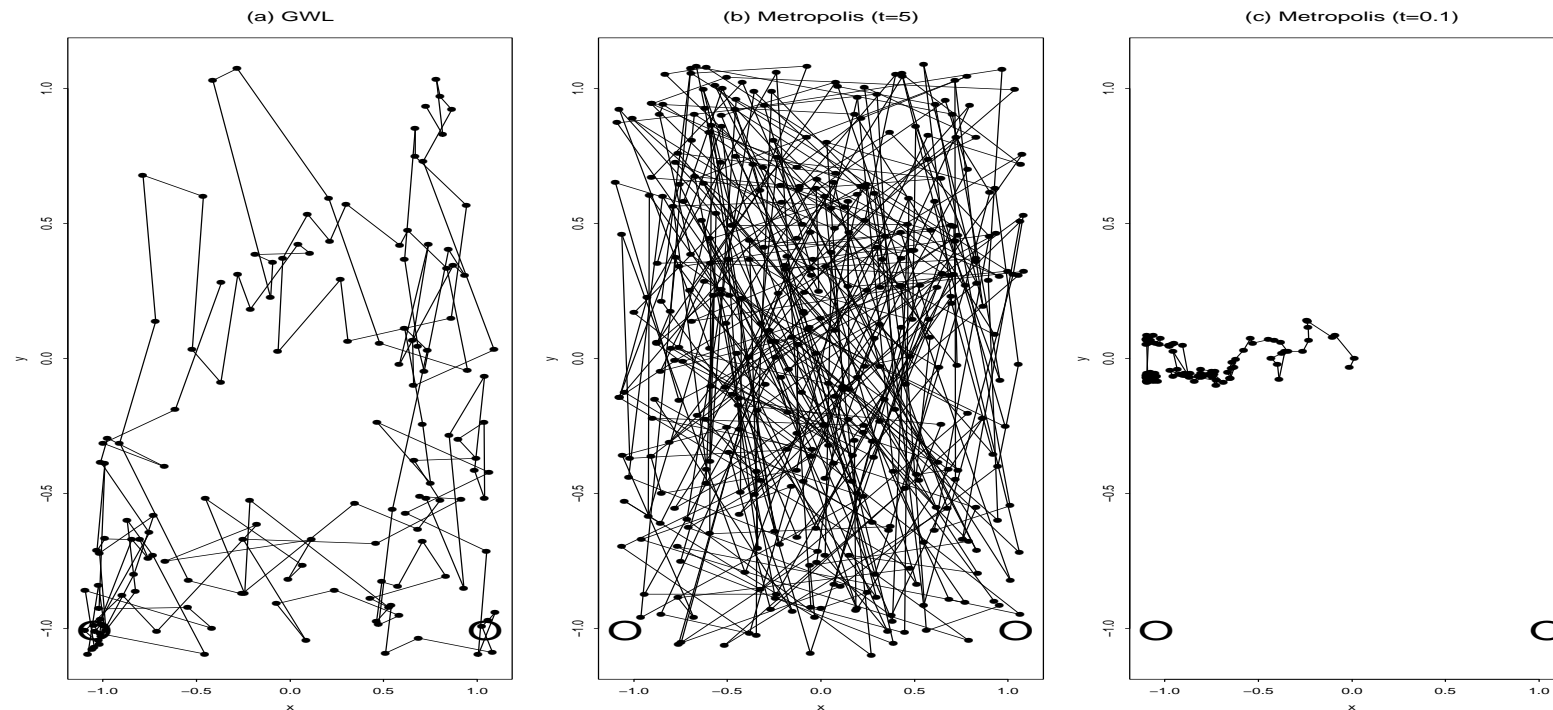


Figure 8: Sample paths of SAMC and the Metropolis-Hastings algorithm. The circles show the global minimum locations. (a) The sample path of a SAMC run. (b) The sample path of a Metropolis-Hastings run at $t = 5$. (c) The sample path of a Metropolis-Hastings run at $t = 0.1$.

Annealing SAMC

The algorithm initiates the search in the entire sample space $\mathcal{X}_0 = \bigcup_{i=1}^m E_i$, and then iteratively searches in the set

$$\mathcal{X}_t = \bigcup_{i=1}^{\varpi(U_{\min}^{(t)} + \aleph)} E_i, \quad t = 1, 2, \dots, \quad (20)$$

where $\varpi(u)$ denotes the index of the subregion that a sample x with energy u belongs to, $U_{\min}^{(t)}$ is the best function value obtained until iteration t , and $\aleph > 0$ is a user specified parameter which determines the broadness of the sample space at each iteration.

Since the sample space shrinks iteration by iteration, the algorithm is called annealing SAMC.

Algorithm	Mean	S.D.	Minimum	Maximum	Succ	Iter($\times 10^6$)	Time
ASAMC	0.620	0.191	0.187	3.23	15	7.07	94m
SAMC	2.727	0.208	1.092	4.089	0	10.0	132m
SA-1	17.485	0.706	9.02	22.06	0	10.0	123m
SA-2	6.433	0.450	3.03	11.02	0	10.0	123m
BFGS	15.50	0.899	10.00	24.00	0	—	3s

Table 6: Comparison of the ASAMC and SA algorithms for the two-spiral example. “Succ” denotes the number of runs (out of 20) found a solution with energy less than 0.2.

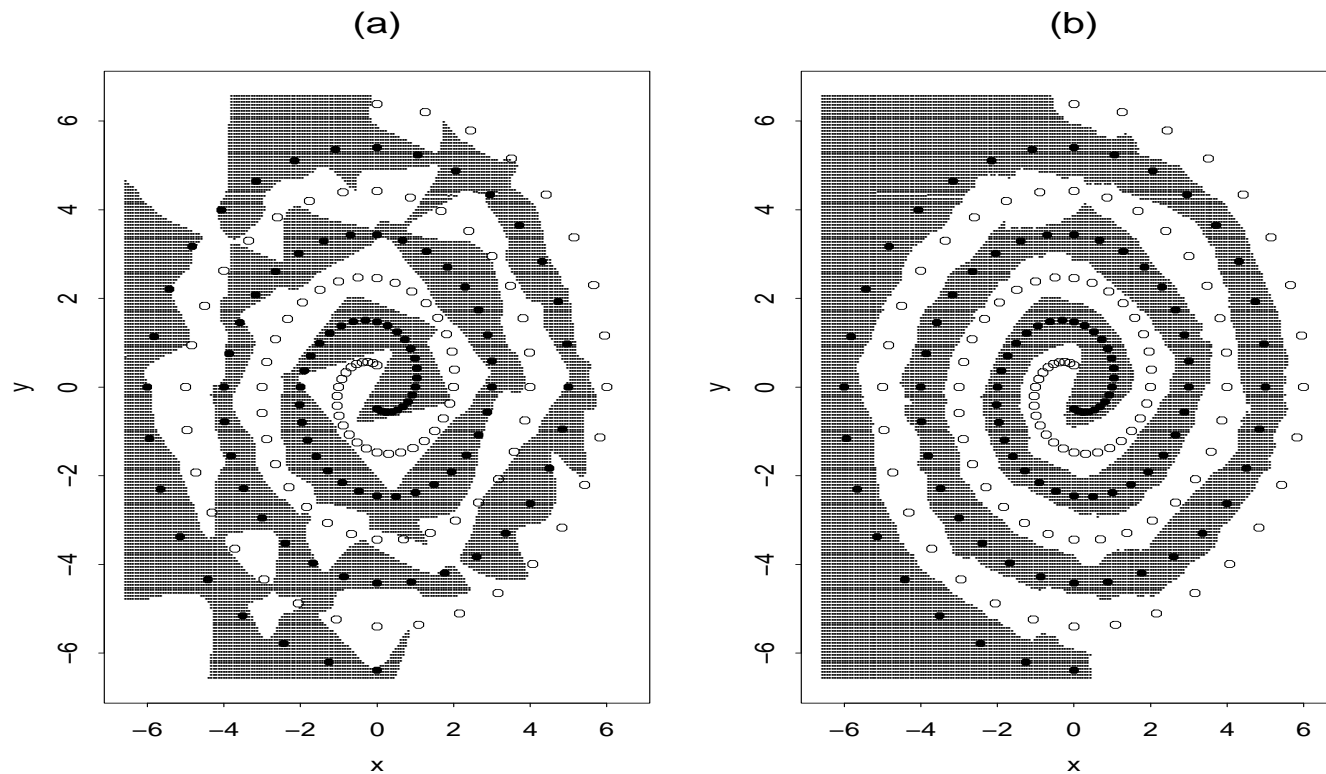


Figure 9: Two-spiral problem: Classification maps learned by a MLP of 30 hidden units. The black and white points show the training data for two different spirals. (a) The classification map learned in one run. (b) The classification map learned in 20 runs.

Advantages of SAMC over simulated annealing

1. *Simulated Annealing: It requires the temperature decrease so lowly, **at the rate** $\frac{1}{\log(t)}$, that it is impossible to be implemented exactly in practice.*
2. *SAMC: The modification factor γ can decrease much faster, **at the rate** $\frac{1}{t}$. In an annealing version of SAMC, X_t will converge in distribution to $f(\mathbf{x})I(x \in \mathcal{E}_\epsilon)$ as $t \rightarrow \infty$, where $\mathcal{E}_\epsilon = \{\mathbf{x} : H(\mathbf{x}) < H_{\min} + \epsilon\}$.*

Further work: convergence rate of annealing SAMC.

Other Applications

- *Importance sampling (Liang et al., 2007, JASA)*
- *Marginal density estimation (Liang, 2007, JCGS)*
- *Normalizing constant estimation (Liang, 2007, Encyclopedia of Artificial Intelligence)*
- *protein folding simulation (Liang, 2004, J. Chem. Phys)*
- *Phylogenetic tree reconstruction (Cheon and Liang, 2007, BioSystems)*
- *Variable selection for high dimensional regression (Liang, Chen and Ibrahim, 2007)*

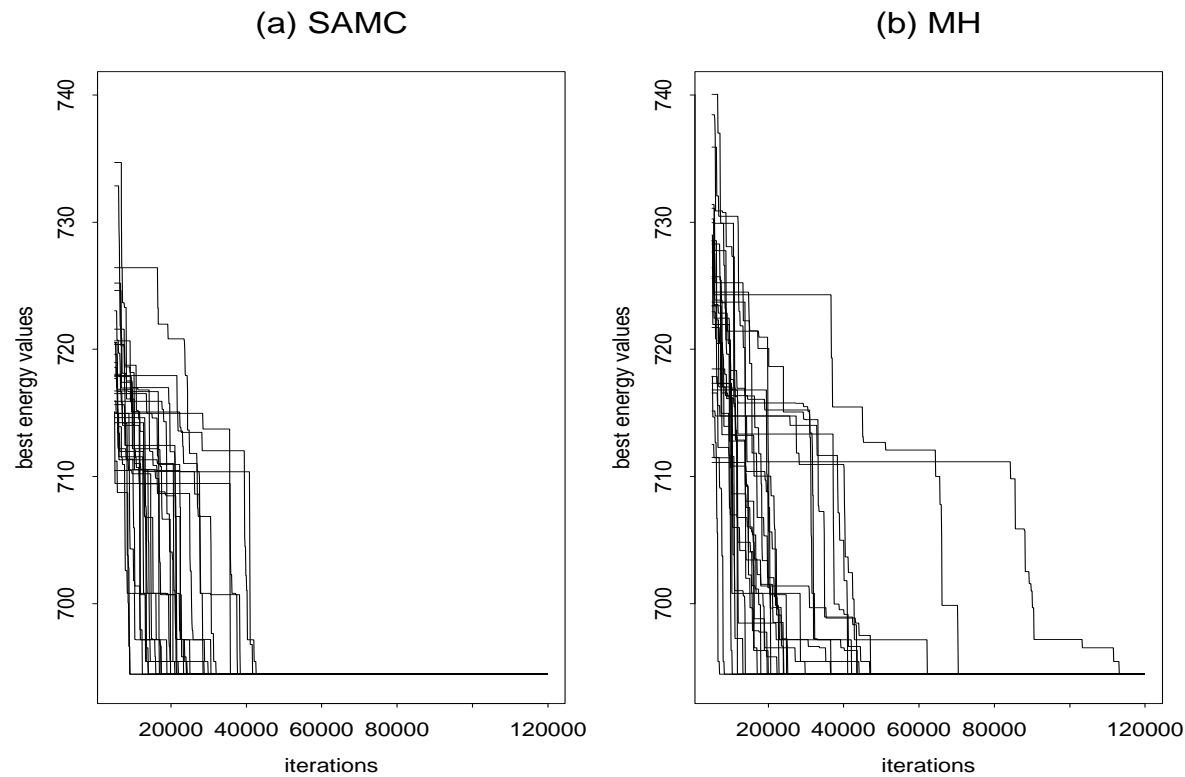


Figure 10: Progression of the best energy values in (a) SAMC and (b) MH runs for a high dimensional regression problem with $n = 150$ and $p = 600$ (Liang et al., 2007).

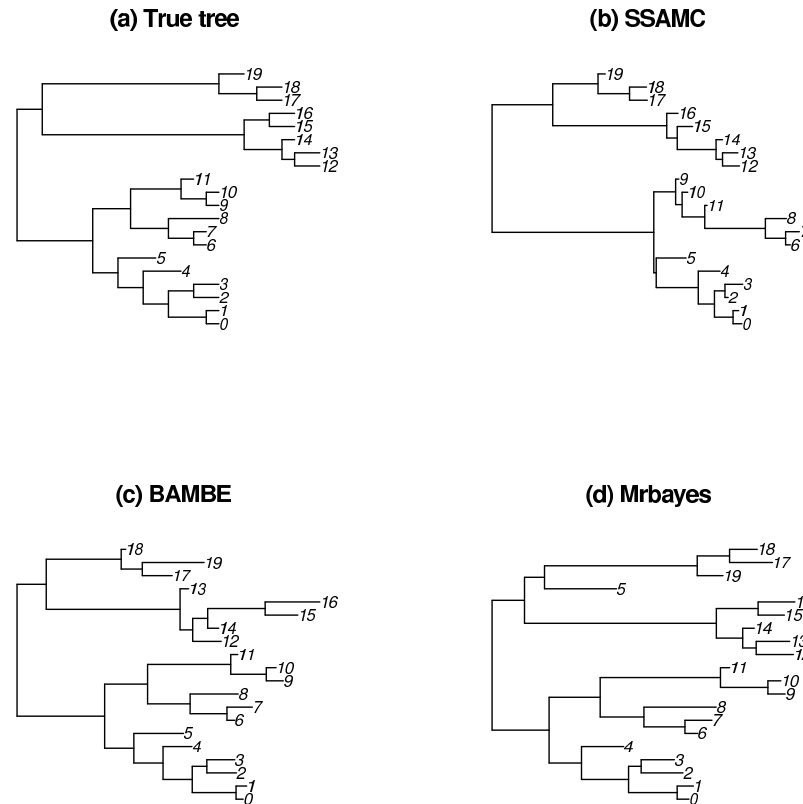


Figure 11: Comparison of the phylogenetic trees produced by SSAMC, BAMBE, and MrBayes for the simulated example. The respective log-likelihood values of the trees are (a) -4209.44, (b)-4196.09, (c) -4197.68, (d) -4198.19.

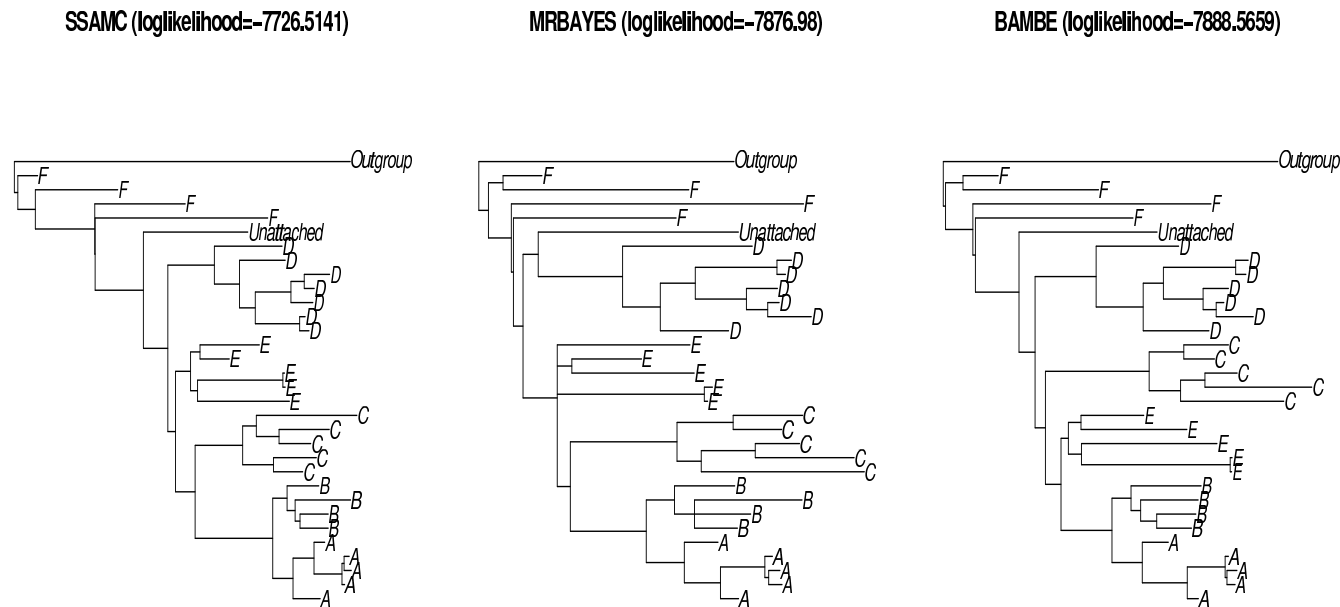


Figure 12: Comparison of the MAP trees produced by SSAMC, MrBayes, and BAMBE for African cichlid fish example. The respective log-likelihood values are -7726.51 , -7876.98 and -7888.57 .

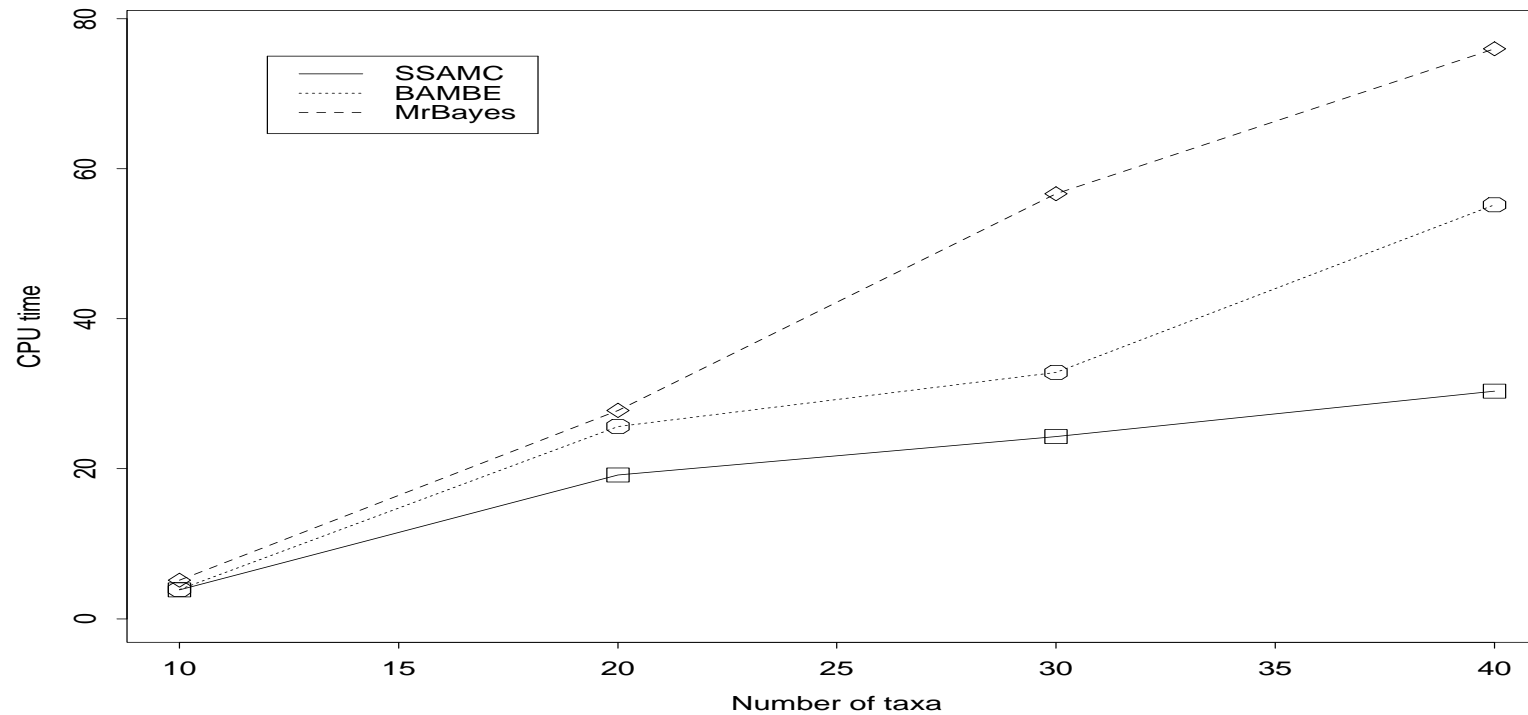


Figure 13: CPU times cost by a single run (2×10^6 iterations) of SSAMC, BAMBE and MrBayes.